

Containers as a Business Service

A decision maker's guide to understanding containers and microservices

Introduction

The responsibility for introducing new technologies that increase business agility and decrease costs are typically the purview of the IT decision maker. Since modern IT has proven to be a constantly changing landscape, it's difficult to keep up. Decision makers need reliable business perspectives on modern technologies and business transformative solutions; containers and microservices, as examples.

Many of the available resources for explaining modern technologies are technical. Decision makers can often learn about technology, but resources that discuss the business impact or perspective may prove harder to find. Containers as a Business Service (CaaS) positions modern IT, containers and microservices, in the context of business considerations.

Contents

1. What you will learn
2. About the author - Bill Bauman
3. An introduction to IT modernization
4. What are containers from a business perspective?
5. What are microservices?
6. How could your organization benefit from containerization?
7. What are the potential drawbacks of containerization?
8. How could your organization benefit from adopting microservices?
9. What are the potential drawbacks of the microservices approach?
10. What about public clouds and containers?
11. What are container orchestration engines?
12. How do you view containers as business services?
13. Why should you monitor modern IT infrastructure with Opsview Monitor?
14. Some final thoughts
15. Next steps?

1. What you will learn

Making informed modern IT decisions

As a business person, whether you've come from a technical background or not, it's important to understand the constantly evolving technology landscape. This whitepaper provides context for the concept of IT modernization, particularly around containers and microservices. While the two are often used in conjunction, they are not explicitly inclusive.

By reading this whitepaper you should gain a solid understanding of how modern IT principles could help your business. This knowledge can assist an informed business decision maker with determining when is the right time to start evaluating a modern IT shift. If you're already in progress, the information within will help you gauge where you are and what's left to do.

This whitepaper is not a reference for an exhaustive list of modern IT technologies. It provides context, examples, and ideas for a business owner and decision maker to understand the framework of how an IT modernization strategy could benefit their organization, from both an IT and business perspective.

2. About the author - Bill Bauman



I am the Principal Product Strategist and Content Lead at Opsview, an infrastructure and applications monitoring software company. I work cross-functionally across the entire organization. It's incredibly fun, but daunting, humbling, and exhausting at times.

After a fun progression of technical, sales and business development roles, I realized I really love telling the story of technology. I used to love technology for technology's sake, but now I love the intersection of humanity and technology. I like to analyze and talk about how we consume technology, and to an extent, how technology consumes us. That brought me to marketing and further on to where I am today.

I really enjoy sales and marketing. I have a great time building business relationships, and I love working with a development team turning feedback into a roadmap for a product that people are absolutely excited to use and that truly makes their professional lives easier and better. I'm also a passionate proponent of DevOps, the opportunity for people in the IT organization to expand beyond their traditional roles and contribute in multifaceted manners to a multitude of teams.

I wrote this piece because I struggled to find a business-oriented explanation of containers in the modern enterprise to share with my friends and colleagues.

3. Introduction

IT Modernization

The trend of IT has always been toward modernization. The current term “IT modernization”, though, largely refers to the transition from traditional on-premises and enterprise computing to more agile, hybrid-cloud (multi-premises), and microservices-based computing. All of this will be explained throughout this whitepaper.

While IT modernization is important for all enterprises, the speed and scope with which it is adopted should be pragmatically decided by the IT business owner within each organization. There is no right way or right time to modernize your IT infrastructure, although the case can be made for the sooner the better.

This whitepaper will serve as a guide to understanding modern IT concepts like containers and microservices, particularly within the realm of business decision making.

Modernizing your IT infrastructure is not a prescriptive solution to success. It is unlikely to make your business suddenly more profitable. It is likely, though, to reduce the amount of time you and your staff spend on traditional administrative tasks. It will free up the technology leaders and IT administrators within your organization to spend more time delivering unique value and technologically differentiated solutions.

Automation, consistency, and continuity are largely the founding principles of modern IT. Your business stands to inherit the benefits of all of these. By automating, there is a reduction in man hours for redundant tasks. Consistency increases the accuracy of deliverables and reduces user error. Continuity reduces management overhead

and the need for you, as a business owner, and decision maker, to oversee the details of every project or be seen as a micromanager while you're really just trying to do a good job. As a leader, you can embrace the intersection of technology and humanity where it starts to directly and jointly address human and business needs.

4. Business overview - containers

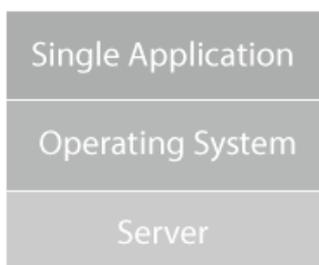
Containers are newly popular, but a mature technology that business decision makers should understand. While containers provide numerous benefits to their traditional counterpart technologies, they don't solve everything, and they have their own limitations.

A container provides a lightweight, isolated environment for applications and their associated dependencies. It is similar to a virtual machine, but with reduced overhead. The reduction in overhead brings with it a reduction in flexibility, though.

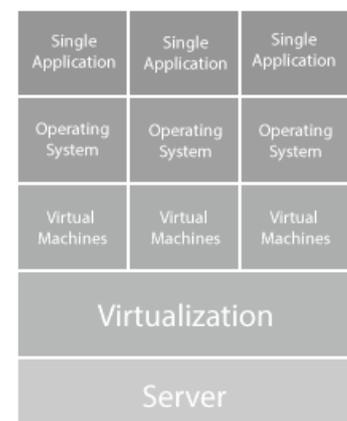
The reason containers have suddenly become so popular has largely been driven by the advent of cloud computing. There is an increasing desire to reduce overhead and time to rapidly deploy new services. By encapsulating the working environment of an application, it adds portability, consistency, and continuity.

Although the impetus for containers has been the cloud, their benefits translate well to the enterprise datacenter.

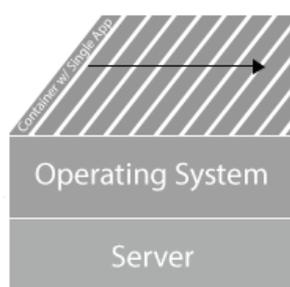
Bare Metal



Virtualization



Containers



5. Business overview - microservices

Microservices, or microservices-based architecture, refers to a development methodology.

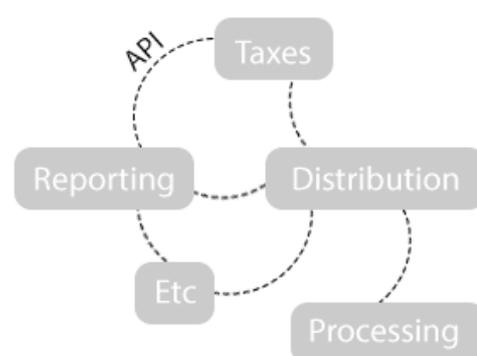
Most traditional applications are considered monolithic. That is, there is typically one, large code base compiled into a large application with all of its functional components inside it. For example, a monitoring application, simplified, would have a user interface front end, a monitoring service, an alerting service, a discovery service, a reporting engine, etc. As a single application, the more functions that are added, the larger the application becomes, the more dependent one section of code becomes on others, and the more difficult it might become to upgrade and troubleshoot throughout the development cycle.

A microservices-based application breaks out all the individual service components of a monolithic application into individual, relatively independent services. The benefits and drawbacks of this approach are discussed further in the microservices section of this whitepaper.

Traditional (Monolithic)
Payroll Application



Microservices-based
Payroll Application



6. Containers - business benefits

The use of containers for business systems can provide significant benefits in terms of time and cost, depending on the current technologies employed within an organization.

If public cloud is part of your IT business strategy, controlling utilization costs is a concern. Since containers are a lighter weight version of application isolation (compared to, say, virtual machines), they can be leveraged to reduce overhead while increasing the density of workloads placed on a public cloud. The same is true for the private datacenter, as well.

Security can be another business benefit of containers. The majority of container technologies deploy containers as immutable images; they can't be modified. This reduces the attack surface for would-be hackers.

The image-based approach to container deployment also makes them portable. In general, a modern Linux container can run on any other Linux host or cloud substrate. The same is true for the portability of Windows containers based on Docker images (the most popular container format to date). Build and run in-house, and move to the cloud when you're ready.

Containers are designed to run a single process each. This makes them ideal to facilitate current, microservices-based infrastructure. Each image is a modern, horizontally-scalable application. Cloud-readiness, cloud-portability and hyperscalability are all inherited.

The modular approach of containers can make solution design and

solution changes much easier. Updating or patching any single component can be limited to a single image in the container layers. This reduces friction for testing, deployment, roll-back, patching, and upgrades. Wholesale, in-place changes of traditional application deployments are no longer necessary.

7. Containers - drawbacks and considerations

Despite the many benefits of containers, there are still business considerations and potential drawbacks.

As with any new technology, it is likely an organization will need to acquire new skills. The cost of skills could be giving time or training to existing employees, or hiring new ones.

Along with new skills, new software will need to be purchased. Many container technologies are available at no charge for development and test, typically through open source distribution models, but it would be unwise for an organization to go to production with unsupported software.

Another consideration for containers is that not all applications are easily put in them. Some legacy applications may require modifications in order to function well in a container environment. If existing staff have a good grasp of containers, they should be able to advise on this. Whether it's existing or new staff, it would be recommended that a container analysis be done to determine the level of effort to containerize existing business applications.

With containers, as with virtual machines and servers before them, there is still the concern of sprawl. Once a technology significantly reduces barriers to deployment, many organizations find themselves deploying a lot more systems than they previously did. Sometimes, these are temporary systems, sometimes, they turn out to be unnecessary. Many of these systems get left behind, left running, and unnecessarily consuming resources.

It is important to have a plan in place to manage potential sprawl. A best practice, both in general and for avoiding sprawl, is to ensure that all systems in an organization have system monitoring applied (see section 13 in this whitepaper). Container Orchestration Engines (COEs), which are discussed in section 11, help limit container sprawl.

8. Microservices - business benefits

Microservices, as discussed earlier in this whitepaper, are applications that have had the functional aspects of their components broken out into smaller, individual programs. The principles behind microservices are largely the backbone of a continuous integration and continuous delivery (CI/CD) process; meaning, the delivery and development cycles do not cease and can iterate quickly.

Deconstructing large applications into small microservice components can make it much easier to scale the application. Instead of having to add physical resources to make a more powerful, single server, components of the application can be distributed across multiple, smaller, virtual machines or servers. Monolithic, or traditional applications, even when horizontally scaled, inefficiently have every service of the app running on all systems, even if only one or two of the app's services need to be scaled.

Because of the scalability of individual services, often times a microservices-based application is more performant overall than its monolithic counterpart. The effect of only starting and scaling the functions you need is less overhead, meaning increased performance and improved user experience.

Since the functional components of a microservices application are all individual, it is relatively easy to upgrade and update each component independently. With some modern applications, you can even in-place update a microservice while the application continues to run. This can provide tremendous flexibility from both business agility and engineering perspectives. Security patches, updates, roll-backs, can all be done faster and more smoothly than with large, monolithic applications.

Troubleshooting microservice components can be easier, as well. With functions separated into relatively small components, they are more manageable. Diagnosing an issue with one microservice doesn't require iterating over the code for all the functions of the application, as it would with a large, monolithic codebase.

9. Microservices - drawbacks and considerations

Despite the many benefits of developing applications in a microservices architecture, there are a few things to keep in mind.

As with many new technologies or methodologies, the initial challenge most organizations will face is finding talented individuals that understand and can implement them. Fortunate organizations may have technologists in-house that have already been pursuing or creating microservices in their existing environment. Others will have to go searching for a growing, but relatively limited pool of existing talent. Retention of in-house talent should also be considered, but that's true at all times.

Microservices-based applications require more planning than a traditional, monolithic app. This is because the functional aspects of the application are split into multiple, individual components (microservices). Just as the functions in a traditional, monolithic application can communicate with each other, the microservices must be able to communicate, as well.

The communication between functions must now be thoughtfully mapped and managed. That's an additional layer of application communication that exists outside simply compiling a program. This message passing layer adds additional complexity in both design as well as initial implementation.

There are many different patterns that can be chosen for the microservices architecture. The choice is often relative to the application's purpose, but beyond the scope of this whitepaper. You might consider **Mulesoft's The Top Six Microservices Patterns eBook**

as a design and developer reference.

Existing applications

A microservices approach might sound like the ideal solution for a significantly challenging, legacy application in an existing environment. While that may be true, it will very likely require a total rewrite of the application. Most microservices applications are written in different languages than their monolithic counterparts. Even if they are in the same programming language, the shift to microservices isn't as simple as just breaking up the code into smaller pieces.

Each microservice must be thoughtfully designed for the new, distributed architecture. The additional communications between the functional microservices may require additional hardware resources. There is often a trade-off between a distributed, scalable approach, and a traditional, monolithic, but perhaps quicker to develop approach. The current and predicted future use of the application should guide this decision.

Troubleshooting is discussed in section 8 of this whitepaper, but it can also be a drawback. It's likely that existing developers are well versed in debugging and troubleshooting traditional, monolithic applications. They may not yet be equipped with the knowledge or toolsets for microservices.

Debugging inter-service communications may require new tools and a new approach. Bear in mind, the modularity of microservices allows them to be run across multiple systems. That's how you achieve horizontal scalability. A monolithic application is traditionally troubleshooted on a single server. Microservices-based applications may be distributed over dozens or even hundreds of servers, containers, etc. Distributing all of the individual functions can sometimes make a bug harder to track down.

10. Cloud container services

Cloud container services and container orchestration bear mentioning because it is likely an organization running containers at scale will be using an orchestrator with them at some point. Cloud container services are about automation. They provide many benefits, although, they may require new skillsets for an organization. In general, cloud container services can:

- Provision physical hosts and containers
- Deploy sets of containers as full services
- Restart and auto-scale containers with a group
- Manage connectivity of containers with other systems

Most public cloud providers now offer some sort of container service. These container services are essentially container orchestration engines (see section 11) implemented to manage and orchestrate containers in a specific public cloud:

- Amazon's Elastic Container Service (ECS)
- Google Container Engine (GCE)
- Microsoft's Azure Container Service (ACS)

Cloud container services make it easier to implement containers without needing speciality COE skills in-house.

Like any cloud offering, cloud container services aren't as flexible as on-premises solutions. They are often easier and faster to get up and running, but they may lock you further into a particular cloud provider's ecosystem.

Cost is determined by provider pricing and workloads.

11. Container orchestration engines

Container orchestration engines (COEs) have the ability to manage some aspects of hardware and are designed to be run in a private datacenter, but are otherwise similar to cloud container services.

COEs can:

- Provision physical hosts and containers
- Deploy sets of containers as full services
- Restart and auto-scale containers with a group
- Manage connectivity of containers with other systems

Popular on-premises COEs include:

- Kubernetes (sometimes known as K8s)
- Docker Swarm
- Mesosphere DC/OS

Kubernetes is perhaps the most popular COE. It was invented by Google to handle the scale of their systems. It has been adapted to work well for container deployments of any size, small to large. It has a bit of a learning curve for staff, but once running, is an excellent scheduler for the containerized services it manages.

Docker Swarm is a more recent entrant into the COE market. While Docker containers are the industry leader, Docker's orchestrator is newer and less mature than other options. It does provide unique options for developers deploying directly to containers.

Mesosphere DC/OS is more than just a COE. DC/OS is capable of managing all of the resources in a datacenter as a single cluster of resources. It can run and manage containers, of course, but it can also run Kubernetes inside it. Though much more complex to set up, for the right purpose, DC/OS can provide benefits beyond traditional COEs.

12. Containers as business services

Containers are a business opportunity, but only if they are approached in an informed, thoughtful manner. Simply deploying container technologies is unlikely to tangibly benefit any organization. Understanding the benefits and drawbacks of containers and microservices, performing due diligence and evaluation of existing applications and infrastructure, and formulating a pragmatic plan to leverage the benefits of modern IT is how an organization will truly see business benefits and reap financial rewards.

Too often an IT business leader wants to dive into the latest technologies to inherit whatever benefits may be there. But without a thoughtful approach, that leader will often miss out on inheriting the business benefits and will merely inherit the overhead and complexity of the new technology.

Bear in mind that containers don't have any special hardware requirements. They run well inside virtual machines, in private datacenters, and on public clouds. So, don't think of containers as an either/or technology. They can provide business benefits regardless of how or where they're deployed.

It's clear that new paradigms are here to stay and that additional paradigms are coming. Many of these new paradigms will include container technologies and microservices-based architectures. Containers aren't the solution for everything, but they can bring cost savings to an organization. It's important to have insight into these costs and cost savings for both existing and new technologies in an organization's datacenter and cloud deployments. To learn more about gaining that insight, see the next section in this whitepaper.

13. Monitoring modern IT infrastructure

The most important aspect of monitoring modern IT infrastructure is that you do it. All systems, traditional or modern, should be monitored. Ideally, a single monitoring tool can consolidate all infrastructure in a unified monitoring platform.

The approach to implementing, and therefore the approach to monitoring modern IT infrastructure, is typically different than before. It requires your team or the monitoring partner of choice understand what mechanisms of action to monitor, what metrics and performance indicators are relevant.

With traditional systems and monolithic applications (which are still critical to monitor), there is a sort of 1-to-1, static approach to servers/VMs and their respective applications.

Modern IT infrastructure may have hundreds of microservices on individual servers. Microservices applications may be spread across many hosts.

As a business person, ensure the vendors and platforms with which you're working have a thorough understanding of modern IT, microservices and containers.

Business insights and automation

Monitoring should provide valuable business insights. There should be a reporting engine to visualize metrics. The metrics should provide observability into systems and events and guide future actions.

There is significant business value in automation, as well. The ability

to automate workloads, and their monitoring, using configuration management tools like Ansible, Chef and Puppet, is imperative to ensuring all systems are properly monitored.

Extensibility, via an inbound, programmatic API, as well as external service connectors, will ensure that the monitoring platform is always integrated and always up to date. This includes existing and emerging technologies that will be implemented throughout a growing IT landscape.

Business more than usual

Opsview Monitor is a monitoring platform that fulfills all of the above business requirements and more. Specifically, the Business Service Monitoring (BSM) capability, as well as Opstacks, are significant differentiators.

BSM defines groups of applications as the service they provide to the business, as opposed to just discrete applications or functions. Notification severity can be defined on a business level, as opposed to just technical observations of online/offline. Reporting based on BSM is presented in a service availability perspective, not system availability. For example, there may have been several critical failures in a given month, but the business service may never have actually been down. Business Service Monitoring provides insights beyond traditional monitoring tools.

Distilled operational intelligence

It's unreasonable to expect operators and systems administrators to be experts in every domain of technology they will be monitoring. The monitoring platform should have some aspects of operational intelligence built in. An awareness of what's being monitored and how to do it is an important aspect of monitoring accuracy and

automation.

Opsview Monitor is shipped with Opspacks. Opspacks contain distilled operational intelligence relevant to the technology being monitored. These are monitoring integration packages that preconfigure Opsview Monitor with best practices and known thresholds for the respective technology they're monitoring, e.g. operating systems like Windows and Linux, databases like Oracle and MySQL, and public clouds like Amazon EC2 and Microsoft Azure.

Instead of adding hosts and configuring thresholds, use Opsview Monitor's Autodiscovery to automatically discover and start monitoring with best practices automatically applied by Opspacks.

14. Some final thoughts

Today's rapid progression of IT is driving new business, not just being driven by the business.

Daily activities often take priority over new technologies and innovation. IT managers and innovators need to be aware of existing systems, roadmaps and commitments to the business, balancing their daily duties with keeping current on modern, innovative technologies. IT modernization like containers and microservices aren't a panacea, but can help relieve pressure. Ensuring visibility and availability of all your systems through monitoring is also critical, regardless of traditional or modern infrastructure.

Stay informed and take a measured approach. Every business moves at its own speed, but typically, IT has to pivot faster and more often than other departments within the business. Trusted technology partners can help guide and implement IT modernization.

The benefits and potential drawbacks of implementing containers and microservices that are presented here are meant to help guide, not be prescriptive. Most organizations can benefit from modern IT and business transformation, if the business is ready and open to incremental change. Some organizations might want to rapidly accelerate their positions in the race for modern IT. If you have questions, please **contact us**, we at Opsview pursue continuous business transformation on a daily basis, as well as help our customers keep a holistic view of their respective IT transformations.

15. Next steps?

Read “The DevOps Manifesto”

Many monitoring tools can expose some aspects of the metrics and data involved in modern IT. It's equally important that the people that sell and support the tool understand traditional, legacy, monolithic applications and datacenter infrastructure, as well as modern IT, microservices, container-based infrastructure.

The monitoring platform of choice should be supported by people who understand modern enterprise business needs, not just technical challenges. Opsview has a passionate and vibrant culture of business and technical monitoring acumen and success.

As organizations evolve, so will the working cultures within them. Most modern IT is being affected by or fully embracing DevOps culture. For a better understanding of DevOps and some inspiration to pursue it, consider downloading **The DevOps Manifesto**, written by the same author as this whitepaper.

Download Opsview Monitor

You can deploy Opsview Monitor free for up to 25 monitored hosts. This gives you access to all our standard features and 60 day access to premium capabilities like business reporting and network analysis.

If you're looking to get started monitoring and increasing visibility immediately, the Opsview Quick Start Professional Services offering is the ideal catalyst to begin increasing uptime and monitoring all your systems with best practices. **Download now.**