Bitnami Vision Whitepaper

The Rise of the Application
Examining Trends in Application Packaging and Deployment

Abstract

Bitnami makes it easier to deploy and manage server applications by providing trusted, easy to use, modern infrastructure stacks across all legacy and cloud-native platforms.

Cloud computing has enabled a new kind of industrial revolution, allowing the automation of what used to be complex, manual tasks performed in traditional enterprise data centers. In parallel, modern software development has evolved to adopt new stacks that contain language runtimes and third-party dependencies not typically included with the base operating system. This 'unbundling' of applications has been accelerated by the rise of containers, which allow the mixing and matching of stack components without depending on the underlying infrastructure vendor. The benefits in the development process have come at an increased operational complexity cost, especially when needing to support multiple clouds, container platforms or hybrid environments.

Bitnami accelerates the adoption of these new platforms by providing ready-to-run solutions that are both easy to use by developers and suitable for production ("Loved by Devs, Trusted by Ops"). Bitnami simplifies the management of multi-cloud, cross-platform environments not by trying to abstract differences away, but by providing functionally equivalent, platform-optimized application and infrastructure stacks for each of them. This allows a degree of standardization across platforms while allowing in-depth integration with vendor-specific features and services. Bitnami addresses the changing application deployment needs of our customers with three core product offerings: an **application catalog**, a curated catalog of pre-packaged, ready-to-run application and infrastructure stacks; a set of **Kubernetes projects** that are driving the application packaging standards; and **Stacksmith**, a tool for simplifying and automating enterprise application migration.

> "Bitnami simplifies the management of multi-cloud, cross-platform environments not by trying to abstract differences away, but by providing functionally equivalent, platform-optimized application and infrastructure stacks for each of them"

The Bitnami application catalog is incredibly popular, having millions of on-premise installations and driving over 1 billion cloud hours of annual usage for our commercial partners, which include AWS, Azure, Oracle, VMWare and Google. The catalog makes it easier to stand up application and infrastructure stacks anywhere, from single VMs for quick casual testing to complex, production-ready clustered setups.

Bitnami is the author or major contributor to key container-based projects such as the Kubeapps application catalog and launcher, the Kubeless serverless solution, and the Helm package manager. As the industry evolves and standardizes around the Kubernetes platform, Bitnami will continue to drive important parts of the Kubernetes ecosystem, focusing primarily around application deployment.

Bitnami Stacksmith is an enterprise-focused offering that enables organizations to migrate their existing, traditional applications to the cloud with minimal effort while providing a seamless path to adopting cloud native practices as appropriate. Applications can be packaged in hours, with little to no changes to source code or configuration files, and they can be deployed to both cloud and container-based platforms such as AWS and Kubernetes. Generated packages follow industry best practices and Stacksmith continuously monitors the dependencies that make up the build, and provides alerts when updates and patches are available. This helps ensure that your applications are always secure and up to date.

This paper aims to explore the major industry trends that have converged to form the current application deployment landscape and how Bitnami fits in it.

Cloud Computing and Automation

Cloud computing was originally promoted as a way to reduce costs by converting capex into opex, and as a way to scale up or down according to usage fluctuations. The reality is that few companies have these scalability requirements. And while cloud cost-savings are real for many scenarios, they are not typically the driving factor for adoption. What the cloud does provide that is really attractive to companies is agility - the ability to significantly reduce the time it takes to launch projects, from months to weeks or even days. This is possible because cloud platforms provide a way to automate every single step of the provisioning of applications, abstracting the underlying infrastructure. There are APIs for everything, from creating databases to launching VMs to performing storage snapshots.

Automation is the key driver for cloud adoption. It allows organizations to replace manual and outdated deployment processes, with programmatic solutions that encapsulate best-practices and can be easily reused across projects and organizational boundaries. Instead of opening a case in a corporate ticketing system and maybe (or maybe not) having access to a bare-bones virtual machine a week later, now with cloud computing, with the click of a button (or an API call) you can launch a fully featured, highly available application environment that incorporates load balancers, databases as a service, monitoring and dozens of other cloud vendor offerings.

Most cloud early-adopters were building greenfield applications, and could do so in a cloud-native way. Best practices for doing so have evolved over time, as the cloud capabilities have evolved themselves, and include clearly separating state from the application with immutable deployments and, more recently, microservices. As cloud is adopted by mainstream enterprises and developers, their initial focus has been on moving existing, traditional three-tier applications to the cloud. The ultimate goal in many cases is to close their existing data centers altogether. But many of these applications cannot easily be rewritten or split into microservices, or the teams in charge do not have the technical capabilities to deal with the added development and deployment complexity.

As the market matures and cloud providers target enterprises performing cloud migrations, new services and tools are emerging to support these traditional workloads. Even container-focused platforms like Kubernetes are adding features aimed towards supporting traditional stateful apps or at least making the transition less painful.

Containers and The New Stack

In the late 90s Red Hat Linux became the standard Linux operating system distribution in corporate environments. At its core, the value in a Linux distribution lies in its ability to curate a subset of freely available software components, configure them to work well together and keep everything up to date, thereby providing a way to upgrade minor versions of the applications and libraries without breaking the rest of the system. The benefits of the industry standardizing around a specific distro are clear: application vendors can write software that targets a specific version of Red Hat and be able to reach the majority of corporate users, which in turn know they can use the same operating system across different hardware providers such as IBM or Dell. This worked quite well for a long time, as most applications were written in C/C++, Java or PHP and did not rely on many third-party dependencies other than, say, the MySQL database or the OpenSSL library.

In the last decade a number of modern language runtimes and development frameworks became mainstream, such as Ruby on Rails, NodeJS and Django. Many of these environments came with a rich ecosystem of third-party libraries with their own package management systems, like Ruby Gems or NPM. Similarly, it is common for applications developed in these languages to rely on a number of third-party server software such as MongoDB, memcached and Elasticsearch. These modern development stacks, such as MEAN (MongoDB, Express, AngularJS and NodeJS) boost developer productivity at the expense of creating significant deployment challenges. A key issue is that all these components evolve quickly and have much shorter release cycles than the traditional multi-year scales of Linux distributions. Users faced the dilemma of having to choose between maintaining their own environments or using the outdated versions bundled with their Linux distribution.

A solution came in the form of containers, a technology popularized by Docker, that allows bundling an application and all of its dependencies in a portable package. Containers effectively 'unbundled' the Linux distribution, allowing users to mix and match components and versions as needed. Containers took the development world by storm and saw rapid adoption among developers due to the significant gains in flexibility and productivity. However, they also brought their own set of new, significant issues. For example, users could now choose from thousands of available applications in the Docker Hub, but had to guess which containers would be actively maintained, could be trusted, and would work well with each other. Deploying containers in production is significantly more complex than the equivalent VM-based setup.

Rise and Fall of Configuration Management

During the 2000s a number of configuration management tools emerged, including the most popular ones: Chef and Puppet. These tools were designed to automate the tasks of installing and configuring software in servers and dealing with different aspects of server orchestration and lifecycle. Initially targeted at bare metal and virtualized deployments, they were quickly adopted for the management of then nascent cloud environments. Configuration management tools became an important part of the DevOps movement, that aims to unite software development and operations.

Though configuration management tools are an important part of the ecosystem, they never fully took off and it is unlikely they ever will. Challenges to widespread adoption included the need to

embrace the tools end-to-end to fully realize the benefits, and the market fragmentation that occurred as Chef, Puppet, Ansible and others fought for market share yet none emerged as a clear market leader. In addition, though in theory configuration management tools allow the encapsulation and reuse of solutions, this has proven difficult not only when sharing across different organizations, but also when sharing between teams within the same company.

Container technology in general, and Kubernetes in particular, will be replacing a lot of existing configuration management tools. Although they are a completely new set of technologies, and there are ways that they can work together, they are also addressing the same core issues. Servers had to be managed because they were difficult and expensive to create and reproduce in case of failure. Containers turn this assumption upside down, doing away with 90% of what's involved in server lifecycle management. Got a new version of your app? Simply build a new container and replace the previous one. Need rolling updates, declarative infrastructure, or built-in high availability? Kubernetes has you covered in those and many other fronts, and has created a rich ecosystem of third-party software without (so far) the level of fragmentation that affected DevOps tools. In many aspects, Kubernetes simplifies complex deployment scenarios by following the opposite approach and clearly separating the Dev from the Ops. The operations team is responsible for the cluster infrastructure itself and provides a clear API-based interface to developers, without having to know what's actually running inside those containers. Though Kubernetes is still evolving rapidly, this approach has gathered an incredible amount of support in a really short time frame.

Many mainstream enterprises are only now moving to the cloud and in the process looking to adopt automation to replace their existing manual procedures and ticketing systems. Most of them will skip the adoption of traditional configuration management systems and instead embrace cloud and container-focused solutions. As infrastructure management gets standardized and commoditized, they will be able to shift their resources towards building and managing applications, the actual reason companies run the infrastructure in the first place!

"Container technology in general, and Kubernetes in particular, will be replacing a lot of existing configuration management tools."

The Bitnami Solution

As described in the previous sections, there are three major, interrelated trends that are fundamentally transforming enterprise application deployment. The first is the widespread adoption of cloud computing and the end-to-end automation it makes possible. The second is the increasing adoption of containers as the preferred way of delivering modern, complex application stacks, replacing the role of traditional operating system distributions. The third is the emergence of container and cloud native platforms and tools like Kubernetes, and its displacement of existing configuration management tools.

These trends solve significant issues but bring new challenges. Gains in developer productivity are accompanied by increased operational complexity, and most traditional organizations are not well-equipped to deal with it. Containers provided by third parties enable choice and flexibility but typically lack the trust, consistency, security and updates provided by traditional operating system vendors. Container and cloud platforms still need to be integrated with the rest of the organization infrastructure and existing application migration is not straightforward.

At Bitnami we have been dealing with these issues since the early days of cloud, helping both end-users and all the major infrastructure vendors. The combination of being 'Switzerland' and our platform-agnostic approach has given us a unique perspective of the industry and allowed us to develop a comprehensive set of tools to address these challenges.

Multi-Cloud, Multi Platform and Everything in Between

All mainstream enterprises have pre-existing, on-premise infrastructure running on bare-metal or virtualized environments. Their challenge is not only finding the best way to leverage the new opportunities that the cloud provides, but also figuring out how to do so without neglecting or disrupting their existing infrastructure investments. As mentioned earlier in this document, the approach of rewriting everything from the ground up to be cloud native is not typically feasible from a financial or organizational perspective. Nor is the approach of treating cloud as an extension of the current data center by applying previous-generation configuration management tools that are not able to fully leverage the capabilities that make cloud attractive in the first place. The problem is compounded with the increasingly common need to target multiple clouds. This can happen either as a conscious organizational choice to avoid lock-in or organically, such as through mergers and acquisitions. Many of the existing tools that aim to solve that problem end up making it worse by trying to abstract the functionality of each of the clouds and provide a consistent high-level interface. They inevitably end up offering the minimum common denominator of functionality and lagging behind the cloud vendor offerings.

Bitnami takes a different approach, focusing instead on providing equivalent infrastructure building blocks as well as pre-packaged applications across all environments and deployment technologies. This provides continuity and consistency across all supported platforms, while at the same time being able to provide deep integration and customization capabilities for each one of them. Users can develop locally on their laptop, test in containers as part of their CI setup, and deploy to their preferred production environment, including VMware, AWS and Kubernetes. Through this entire process they can use the same version of their language runtime or backend database, optimized for each platform and kept secure and up-to-date by Bitnami. During development they can use a native binary or lightweight Docker compose setup of say, a Rails environment with a MongoDB backend, and choose to deploy to Amazon's cloud using a production-ready, multi-tier cloud formation template that follows AWS best practices. The bits in each one of the deployment artifacts are basically the same, configured and optimized by Bitnami. Making the deployment platform a late-bound decision has some other benefits such as the ability to choose at any given time whether an app should run on-premise or elsewhere, and reducing the inherent complexity of supporting multiple clouds.

Kubernetes is the new Linux

Over the last two decades Linux has become the preferred deployment platform for server software in a large number of commercial organizations. Hardware vendors, ISVs and companies

standardized around the particular set of software versions and specific configurations bundled with each major version of Red Hat Enterprise Linux, the leader in the space. However, as application stacks became more complex and the speed in which the underlying components evolved increased, the Linux distribution vendors were not able to keep pace. Container technology came to the rescue, with the side-effect that they 'unbundled' the applications from the underlying operating system infrastructure. The Kubernetes project took this further, providing a platform for production deployments that was quickly adopted by the industry. Kubernetes provides a solid foundation for further standardization of other important aspects of application deployment and management such as configuration, monitoring, troubleshooting, updates and security.

Bitnami has embraced Kubernetes and provides important contributions related to application deployment. Kubeapps makes it easier to discover and launch Kubernetes applications packaged as Helm charts, a project that we are major contributors to. Kubeless is a Kubernetes-native serverless implementation that is lightweight and seeing rapid uptake. Our Kubernetes best-practices runtime framework extends the vanilla Kubernetes environment provided by infrastructure vendors, providing a ready-to-run setup that includes monitoring and ingress and credential management. These and other projects that we are authors of or contributors to have the common goal of simplifying the usage of Kubernetes for mainstream application deployment, allowing customers to bridge the gap between their needs and what's possible with current technology. For us, Kubernetes is another platform (albeit a very exciting one!) in which we provide the same value we already do for others: trusted, optimized, up-to-date content that simplifies development and production and the tools to make it easily for organizations to adopt to package and deploy their own applications.

Lowering Adoption Barriers

Most successful business are built over the course of years, and their systems reflect this. Today's shiny technology will be tomorrow's legacy systems. At Bitnami, we put a lot of effort in helping our users improve the way they create and manage their applications without forcing them to throw away their existing infrastructure and start from scratch. Part of the reason why the new crop of container-focused solutions has not seen commercial success in the marketplace is that most of them force companies to embrace the technology completely. This means that customers can only use these tools for new projects, or that they need to spend time rewriting their existing solutions so they can be containerized. While this may make sense for some subset of applications, for many, it will not: if the original application development team has left the company or moved on to other projects, for example, or when the ROI of a rewrite is simply not there. The problem is compounded by having to maintain complete separate solutions for the traditional VM-based workloads and the new container-based solutions.

Bitnami Stacksmith solves all these problems elegantly. It turns existing, traditional server applications into production-ready, platform optimized cloud images, complete with the necessary deployment templates ready for your chosen platform. And all with minimal to no changes to the source code. This allows enterprises to take care of migrating applications to the cloud quickly and effectively. Stacksmith can integrate with existing configuration management tools that may be already in place, but does not require them - in many instances, it can replace their functionality at a fraction of the complexity. Stacksmith creates deployment templates and artifacts that

encapsulate industry best practices, and it keeps track of all the bundled library dependencies and components. As updates are available due to new versions or security issues, Stacksmith notifies users and can trigger a rebuild automatically or on demand.

For new application development or applications that can benefit from containerization, Stacksmith provides Kubernetes support that is seamlessly integrated with the rest of VM-based platforms. In other words, Stacksmith meets customers where they are while providing a consistent path to the future.

Conclusion

Cloud computing and containerization technologies are here to stay and are having a profound impact in how organizations build, deploy and manage their applications. The benefits of automation are often accompanied by additional complexity in other aspects of application delivery that are not addressed by existing tools, designed for a previous era. As a result, IT organizations face an increase in the number of platforms and technology stacks that they need to support while simultaneously being expected to reduce cost and delivery times. Bitnami provides platform-agnostic solutions for these challenges that are loved by developers and trusted by operation teams. We do so by providing a popular catalog of applications and infrastructure components that are optimized for each platform that we support, as well as the Stacksmith tool that enables enterprises to package and deploy their own applications and maintain them secure and up to date.