SPK
and Associates

Case Study: **Using DevOps and Cloud to rapidly respond and scale**

Background

SPK was contacted by a software company that provides healthcare cost optimization through a web platform. Their mission is to provide customers with a method to reduce spending while maintaining quality healthcare for their employees.

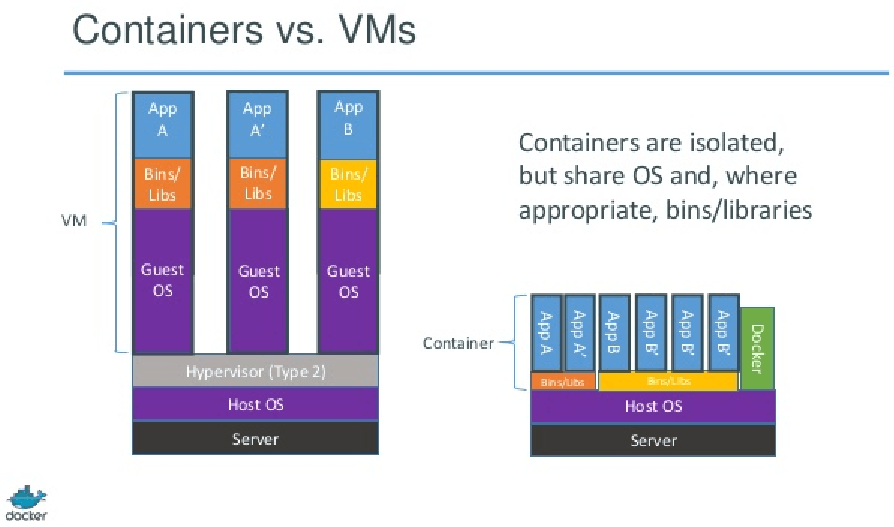
With an expanding number of choices of health providers, plans, and costs, the average company needs to invest a significant amount of time and energy in order to understand these options. They must decide on both provider and plan to meet the increasingly diverse healthcare requirements of their workforce. This application gives growing businesses a data-driven benefits platform to make the best plan decisions.

Development and Pre-production

The client had a seasoned engineering team with exceptional talent. As they developed their product, they realized that their platform needed to address some fundamental challenges:

- Healthcare plans are constantly changing.
- Their customers who need these plans have a diverse range of requirements.
- Their customers would need a reliable interface to access their healthcare information
- The salesforce needed their own instances in order to demonstrate and test plan suggestions for their customers.
- HIPAA compliance dictates a strict list of requirements regarding security and the storage of PHI.

The engineering team quickly decided that Jenkins and Docker would provide a foolproof and efficient method to build and deploy releases. Out of each Jenkins build came a Docker container. A Docker container is essentially a copy of everything needed to run a set of one or more applications inside one file. Here's a graphic from Docker's website to show how it is different from a Virtual Machine (VM):



Because the OS is independent of the container, these containers are very streamlined and have much lower system overhead than having a virtual machine for each. The largest benefits of Docker containers are:

- **Portability –** An image can be moved from one host to another without determining if the "right version of Apache" is there or the "right version of Nginx" is present. This proved useful as the customer moved to AWS and the EC2 environments.
- **Version control –** each image has a version that is easily identified post-install
- **Lightweight deployment -** minimal levels of code are transferred during deployment.

Amazon AWS

The customer realized that it needed scalable resources, reliability, and flexibility. They turned to Amazon AWS to allow growth of their environment at a cost effective price point. Using the Cloud allowed them to meet some of their initial requirements:

- Existing customers need a **reliable interface** to access healthcare information
- The salesforce needed their **own platform instances** in order to demonstrate and test plan suggestions for their customers.

Customers – a blessing and a burden

As the company prepared for customers coming onboard, new challenges emerged. Additional environments were needed to ensure releases met baseline needs before being used by customers. Also, backups started to become a challenge as the database size increased to where a simple export would not meet the service level agreements of their customers.

The company also needed a way to test on a platform that didn't change by the minute. With more customers, manual testing and deployment of new releases over varying platforms would become untenable.

Maintenance would become an issue as well. Since Docker images were being pushed manually, a short downtime was seen when stopping an existing container, loading, and starting the new container.

New Problems to be solved:

- Deployment time would increase due to complexity of multiple environments and a single person manually handling the releases
- Specialized requirements would increase as new customers with diverse needs signed up
- The code base was growing quickly. This increased development complexity where changing one line of code would increasingly impact other areas
- The AI engine would need more compute power to support the growing customer population
- Backups of sensitive data needed to happen reliably without impact to production
- New HIPAA requirements were identified requiring higher security

SPK assistance arrives!

The company reached out to SPK to help create a DevOps process and Production Cloud architecture which could address these pain points. SPK's years of experience with Cloud technology, engineering, and compliance regulations were well suited to help solve their specific needs.

Step 1: **Set up a private Docker repository.** Due to the sensitive nature of the application and the data, the public Docker Hub could not be used in Production. Jenkins was then incorporated so that once a build completed, a new Docker image would be **pushed automatically** to the private repository.

> *With more customers, manual testing and deployment of new releases would become untenable.*

Step 2: **Address HIPAA compliance requirements.** HIPAA states that data must be encrypted at rest as well as in transit. In addition, servers could not be shared with the general public. Think of Heartbleed, whereby an attack could leak memory contents from the server.

SPK addressed this by moving instances to dedicated resources on AWS. In addition, internal connections were also moved to SSL, and encrypted storage volumes were implemented.

Step 3: **Increase database reliability.** SPK tackled this issue by leveraging Postgres streaming replication. In addition, backups became more robust, and were sent to S3 for retention.

Step 4: **Implement monitoring.** This was a crucial step. With the application in production, there was no visibility when something broke. By implementing Nagios, SPK provided that much needed visibility into the environment. Disk usage issues, database issues, crashed machines, etc. would alert the Operations team before the customers experienced a problem.

Step 5: **Tighten security.** This required a multipronged approach. Security groups needed a narrower scope. Operating system services needed to be pared down. Passwords needed to be updated. SSH keys needed cleansing. There is no comprehensive list – security is an ongoing, iterative process.

The end result:

- Full automation of builds in Jenkins.
- Upon successful compilation, builds are automatically sent to a Docker repository, saving the company 8 hours per week, and allowing them to increase releases by 400%.
- Replacing a server environment would take minutes instead of hours.
- DEV, TEST, QA, and PROD are updated to a new release with a single command
- A fully redundant Production environment on a cloud infrastructure.
- A Docker registry of all objects on a S3 data store. This allows both rollback and roll forward of releases with a single command. It also ensures no releases are lost.
- Overall, a more robust, secure operating environment that saves the company time, so that they can focus on their core strength – their product.

About SPK and Associates

We help our clients build ***amazing*** collaborative product development environments, supplying needed expertise in Engineering process and applications, DevOps and Cloud, and IoT Design and Operations. We are a technology solutions partner with deep industry knowledge in Medical Device, High Tech, and the Financial industries.

For over 20 years, SPK has helped clients worldwide design, build, and release products better, faster, and more cost effectively. We excel at improving time to market and product quality in regulated industries, such as Medical Device. We also manage our clients' customer-facing Cloud environments, with exceptionally high standards.

Replacing a server environment would take minutes instead of hours.

"Builds are now automatically sent to a Docker repository, saving the company 8 hours per week, and allowing them to increase releases by 400%."