Case Study

haku Deploys Faster on Ruby on Rails with Docker's Synchronized File Shares

About company: haku provides an all-in-one platform for large endurance and non-profits, with features ranging fromevent planning and provides capabilities from registration and customerparticipant engagement to fundraisinganalytics and e-commerce.
Industry: Endurance Events and Non-Profits
Location: Miami, Florida

Highlights

- **Days-to-hours reduction in setup time** for new environments.
- **10x improvement in local response times** with synchronized file share for Mac users.
- **Better environment consistency**, faster onboarding, and overall development enhancement.

"With our sized code bases, the native bind mounts, a limitation of the OS was limiting Docker, were just not cutting it. They were much too slow for us. Synchronized file shares and its integration into Docker Desktop changed the game for us, providing a tenfold improvement in response times locally."

Steven Witte
Senior Software Engineer, haku

Table of Contents

Introduction

In 2013, haku was founded in Miami, Florida, by brothers Carlos and Jose Escobar, with the vision to revolutionize the management of endurance events and non-profit fundraising like marathons and triathlons. Getting their start in the endurance event space, their first mission was to simplify event organization, offering a platform that manages the intricacies involved for both event organizers and participants.

Within the endurance event management industry, where seamlessly managing large-scale events is needed, haku focused on solutions for enterprise organizations like the Bank of America Chicago Marathon, Marine Corps Marathon Organization, and the Eagles Autism Foundation, which host up to 30,000 participants. The complexity of each event demanded flawless execution, exceptional participant engagement, and rigorous logistical coordination. haku's commitment to leveraging cutting-edge technology set them apart, promising a level of service and efficiency that would redefine industry standards.

Early in their journey, haku recognized a critical barrier to their growth and innovation: the complex and time-consuming process of setting up development environments. This internal challenge was multiplied by file synchronization challenges in MacOS development, threatening to slow their development teams and their response to market demands in endurance event management.

haku’s challenges mirrored broader struggles within the tech industry: ensuring consistent performance across varied operating environments and a quality developer experience from onboarding to tenured lead. The search for a solution led them to Docker and Mutagen, a Docker Extension now integrated into Docker Desktop as Synchronized file shares. These technologies promised to address specific scaling issues fundamental to their development lifecycle. This decision marked the beginning of a pivotal chapter in haku's story that would test their commitment to innovation and ability to harness technology to pursue their vision.

The Problem

Slow setup in MacOS development

The initial setup of developer environments at haku was cumbersome and time-intensive, mired in a complex web of dependencies. This setup was crucial for onboarding new talent and maintaining the pace of innovation. Before Docker, setup took one to three days, slowed by environment-specific technical complexities. For a company at the forefront of technology in endurance event management, this delay was more than just an inconvenience; it was a barrier to agility and rapid development.

Each developer's machine required meticulous configuration of numerous software dependencies. This process was fraught with potential errors and inconsistencies, leading to what is commonly known as "dependency hell." The dynamic nature of software development meant that these dependencies were continually evolving, further complicating the setup process.

haku's development team uses exclusively Mac computers. These team members faced persistent challenges with file synchronization, a critical aspect of their workflow involving frequent exchanges of updated code and files between their local machines and development containers. Steven Witte, Senior Software Engineer, shares, "With our sized code bases, the native bind mounts (a limitation of the underlying OS) in Docker were just not cutting it. They were much too slow for us." The severe degree to which the developers were affected caused significant frustration.

Realizing these technical challenges impeded haku's ability to innovate and respond to market demands catalyzed change. The leadership team, recognizing the critical need for a solution to streamline the developer environment setup and enhance productivity, turned their focus toward Docker. This strategic decision was poised to address the specific challenges faced by their team and unlock new levels of efficiency and innovation.

> "We've seen big performance improvements across the board with the new built-in synchronized file shares with Docker Pro."
>
> **Adam Toro**
> VP of Engineering, haku

> "The transition to Docker, especially for our development environments, has made onboarding new team members significantly faster, reducing the setup time from days to hours."
>
> **Adam Toro**
> VP of Engineering, haku

> "Docker Desktop's synchronized file share feature, was a no-brainer for us. It made local development so much more efficient."
>
> **Steven Witte**
> Senior Software Engineer, haku

The Solution

Embracing Docker and improved file synchronization for enhanced efficiency

The turning point for haku came with the strategic adoption of Docker. This platform streamlined the setup and management of their Ruby on Rails tech stack across various environments. However, Docker's acquisition and integration of Mutagen into Docker Desktop, particularly for Mac users in haku's team, marked a significant leap in their technological capabilities. "We've seen big performance improvements across the board, with the new built-in synchronized file shared with Docker," notes Adam Toro, VP of Engineering.

Docker's containerization technology offered a consistent and isolated environment for development, testing, and production, eliminating the variability plaguing their previous setup. The integration of synchronized file shares into Docker Desktop was the solution to the file synchronization challenge—precisely the slow performance issues faced by developers using Macs—was instrumental. Witte emphasizes this importance: "Docker Desktop's synchronized file share feature was a no-brainer for us. It made local development so much more efficient." The dramatic improvement in speed and reliability of file sharing between host and container environments ensured that changes made by developers were quickly reflected in the application without the need for tedious manual synchronization.

> "We've seen Docker not just as a tool for development but as a significant enhancement to our entire development cycle, making things faster and more reliable."
>
> **Adam Toro**
> VP of Engineering, haku

> "Absolutely. Docker's impact has been all-encompassing, from development to deployment, ensuring consistency and speed."
>
> **Steven Witte**
> Senior Software Engineer, haku

> "By moving to Docker, we've upgraded our tech stack with much less hassle, making it easier to adopt the latest software versions and reduce dependency conflicts."
>
> **David Lozano**
> Senior DevOps Engineer, haku

Key Benefits

Rapid machine setup and reduced onboarding time

The transition to Docker has drastically reduced the time required to set up new machines for development. Toro shares, "The transition to Docker, especially for our development environments, has made onboarding new team members significantly faster, reducing the setup time from days to hours."

Consistency across environments

Docker ensures that all dependencies, operating system versions, and libraries are consistent across all environments. This consistency solves the problem of discrepancies arising due to differences in environments, such as between local macOS machines and Linux production servers.

Upgrading tech stack and running multiple versions of Ruby on Rails

Docker has been pivotal in upgrading Haku's tech stack, allowing them to run different versions of Ruby on Rails more efficiently. This capability has enabled the development of new apps on the latest versions of Ruby on Rails, circumventing the challenges associated with dependency management on their macOS systems.

Seamless introduction of new services

Docker's use has streamlined the process of introducing new services into Haku's platform, allowing for a seamless and less labor-intensive experience. This efficiency comes from the ease of setting up development environments with Docker, ensuring consistency and reducing the complexities associated with managing multiple versions of software across different developers' machines.

Development and shipping speed

Utilizing Docker has allowed Haku to develop and ship updates faster. The consistent environments between development, staging, and production eliminate surprises during deployment, ensuring that what has been tested in lower environments performs identically in production. “We’ve seen Docker not just as a tool for development but as a significant enhancement to our entire development cycle, making things faster and more reliable."

Outcomes

The finish line: containerization a model for technological evolution

Implementing Docker Team brought about transformative outcomes for haku, with the most notable impact being the drastic reduction in environment setup times. What previously took days was now accomplished in a matter of hours, a change that significantly accelerated the development cycle and enhanced productivity. The benefits extended beyond setup efficiency; application performance saw substantial improvements, with faster load times and a more seamless development experience. “Synchronized file shares and its integration into Docker Desktop changed the game for us, providing a tenfold improvement in response times locally”, says Witte.

Moreover, integrating these technologies facilitated a more agile and responsive approach to development. The ability to quickly spin up or tear down environments as needed allowed haku to experiment more freely and iterate rapidly, ensuring that they could stay ahead of the curve in an industry where technological agility is a key differentiator.

The case of haku exemplifies the impact of targeted technological interventions on an organization's operational efficiency and innovation capacity. By adopting Docker, haku addressed specific challenges and improved its internal processes. They positioned themselves for future growth and success in the endurance event management industry.

This journey highlights the importance of continuous technological evolution and the strategic implementation of tools to enhance development processes, reduce time-to-market, and ultimately drive business growth. haku's experience is an inspiring blueprint for organizations leveraging technology to overcome operational challenges and achieve their business objectives.

> "Docker has eliminated 'dependency hell' for our development team, streamlining our workflows and significantly reducing setup times for new developers."
>
> **Steven Witte**
> Senior Software Engineer, haku

> "Docker has enabled us to mirror our production environment locally, eliminating surprises during deployment and enhancing our service delivery to Haku customers."
>
> **David Lozano**
> Senior DevOps Engineer, haku

> "Thanks to Docker, we're now able to develop and ship faster, with the confidence that what we test in our lower environments will perform the same in production."
>
> **Steven Witte**
> Senior Software Engineer, haku

Find a subscription that’s right for you

Contact an expert today to find the perfect balance of collaboration, security, and support with a Docker subscription.

Contact Sales