How Docker broke in half

The game changing container company is a shell of its former self. What happened to one of the hottest enterprise technology businesses of the cloud era?

Scott Carey

UK Group Editor, InfoWorld

Sep 7, 2021 3:00 am PDT

By

tom.ruethai / Shutterstock / Docker

Docker didn’t invent containers—the method of packaging up computer code into compact units that could be easily ported from laptop to server to server—but it did take them mainstream by creating a common set of open source tooling and reusable images which suddenly allowed all developers to build their software once and run it anywhere.

The ease with which Docker enabled developers to “containerize” their code and move it from system to system quickly established it as something of an industry standard, upending the dominant method of deploying applications on virtual machines (VMs) and establishing Docker as one of the fastest adopted enterprise technologies of a generation.

Today, Docker is still alive, but it is a fraction of the company it might have become, having never succeeded in turning this technological innovation into a sustainable business model, eventually leading to the sale of its enterprise business to Mirantis in November 2019. InfoWorld spoke to more than a dozen former and current Docker employees, open source contributors, customers, and industry analysts to hear the story of how Docker broke into pieces.

Docker is born

Founded as DotCloud in 2008 by Solomon Hykes in Paris, the company that would become Docker was initially designed as a platform as a service (PaaS) for developers to easily build and ship their applications.

Hykes was soon joined by his friend and fellow programmer Sebastien Pahl, before moving to Silicon Valley together to go through the prestigious Y Combinator program in the summer of 2010. Having

already been rejected once, Hykes and Pahl reapplied, with Pahl's dad fronting them the money for plane tickets to San Francisco a few weeks ahead of their interview. Alas the pair were rejected again, until YC alumnus James Lindenbaum, the founder of a competing company called Heroku, stepped in to vouch for them.

Docker as we know it was first demoed by Hykes at PyCon in March 2013, where he explained that developers kept asking for access to the underlying technology powering the DotCloud platform. "We did always think it would be cool to be able to say yes, here is our low-level piece, now you can do Linux containers with us and go do whatever you want, go build your platform, so that's what we are doing," he said during that talk.

"It sounds corny, but Solomon and I were talking pre-release and we could see all the container ships coming into the port of Oakland and we were talking about the value of the container on the world of shipping," Ben Golub, Docker CEO between 2013 and 2017, told InfoWorld. "The fact it was easier to ship a car from one side of the world than to take an app from one server to another, that seemed like a problem ripe for solving."

The Docker open source project quickly built up steam, attracting thousands of users, high-profile partnerships with the likes of Microsoft, AWS, and IBM, and wheelbarrows full of venture capital money, including early investments from Peter Fenton at Benchmark and Dan Scholnick at Trinity Ventures. The refocused company changed its name to Docker and went on to raise nearly $300 million from the likes of Benchmark, Coatue Management, Goldman Sachs, and Greylock Partners. However, like many open source software-based companies, it struggled to find a profitable business model and those investors never got their big exit.

“Solomon built one of the most compelling technologies of the past 20 years and in the business of packaging something up with an opinion and making it extremely valuable to a huge number of developers, Docker was massive,” RedMonk analyst James Governor said. “Did Docker make bad decisions? Clearly yes, but the venture capitalists went mad and the amount of money they threw at them meant it must have felt like they could do anything, which was problematic.”

Fast forward to 2021 and the short version of this story is that the massively popular open source container orchestration tool Kubernetes ate the lunch of Docker (the business) by displacing its primary profit center: an enterprise version of its own container orchestration tool called Docker Swarm. However, the true story is far more complex.

Commercializing open source is hard

The combination of huge amounts of venture funding, a quickly growing competitive landscape, and the looming shadow of cloud industry giants all wanting a piece of the pie created a pressure cooker environment for the young company to operate within.

“There’s a saying that ‘when elephants fight, the grass gets trampled,’ and it became clear to us this wasn’t just about Docker, but how the cloud vendors were competing with each other. They all wanted to pull us in different directions. It was a constant juggling act to stay true to our values and roots and build a business,” Golub said.

The former CEO notes that all of these factors created “natural tensions” as Docker grew. “We wanted to build great community and monetize the developer product, while also building a great operator product to allow customers to build and deploy containers at scale," Golub said. "That was the vision and pretty quickly we realized we had to scale rapidly and didn't have a lot of time to balance the community and being a commercial business ... at a startup you are making 100 decisions a day, and you hope 80 are right."

Docker started to get serious about a business strategy to monetize its leading position in the container world around 2014, when the company spent some of that VC money on the acquisitions of Koality in 2014 and Tutum in 2015, while also launching the first iteration of its own enterprise support program.

These investments led to products like Docker Hub—which you can think of a bit like a GitHub for Docker images (which also exists now)—and eventually Docker Enterprise. But none of these products truly took off with enterprise customers, who were generally happy to work with more established partners, or build rather than buy solutions, as Docker labored to produce a set of products customers truly wanted.

"We never shipped a great commercial product," Hykes told InfoWorld while on vacation in France this summer. "The reason for that is we didn't focus. We tried to do a little bit of everything. It's hard enough to maintain the growth of your developer community and build one great commercial product, let alone three or four, and it is impossible to do both, but that's what we tried to do and we spent an enormous amount of money doing it."

"There was zero technical delivery happening outside of open source," Nick Stinemates, former vice president of business development and technical alliances and one of the earliest employees at Docker, said. "There was a fundamental inability to deliver commercial software."

With the benefit of hindsight, Hykes believes that Docker should have spent less time shipping products and more time listening to customers. “I would have held off rushing to scale a commercial product and invested more in collecting insight from our community and building a team dedicated to understanding their commercial needs,” Hykes said. “We had a window in 2014, which was an inflection point and we felt like we couldn’t wait, but I think we had the luxury of waiting more than we realized.”

Others think Docker gave too much away for free too early on. “They put something out for free that nailed it, home run,” Google’s Kelsey Hightower told Increment magazine earlier this year. “They solved the whole problem and hit the ceiling of that problem: Create an image, build it, store it somewhere, and then run it. What else is there to do?”

Hykes disagrees with this assessment. “I think that is wrong and generally speaking the core open source product created massive growth which created the opportunity to monetize in the first place,” he said. “Lots of companies monetize Docker successfully, just not Docker. There was plenty to monetize, just Docker failed to execute on monetizing it.”

For example, both Red Hat and Pivotal (now part of VMware) were early partners with Docker, integrating Docker containers into their commercial PaaS products (OpenShift and Cloud Foundry respectively) and contributing back to the open source project.

“If I am being generous, the contributions from Red Hat early on spun Solomon out a bit,” Stinemates said. “Solomon burned a lot of bridges and there are threads on Hacker News of him starting fights with naysayers. Enterprise partners couldn’t have this with Solomon.”

Today, Hykes says that he was guilty of confusing "community with ecosystem." Red Hat specifically "weren't part of the community, they never rooted for the success of the Docker," he said. "The mistake on our end was desperately wanting them to be part of the community. In retrospect we would never have benefited from that partnership."

As a result, early customers like the travel tech company Amadeus turned to Red Hat in 2015 to fill what they saw as an enterprise-grade void left by Docker. "We transitioned directly from a pioneer mode, where we were leveraging the open source versions [of Docker], to a strong partnership with Red Hat, where they were covering the support of container tech for us," Edouard Hubin, head of cloud platform solutions at Amadeus, told InfoWorld, via email. "Containerization was the first step of the technological change away from virtualization. The real game changer for the enterprise was the container orchestration solution. Clearly Docker lost this battle to Kubernetes and that was a very difficult situation for them."

The Kubernetes decision

Docker would come to rue an earlier set of decisions surrounding its refusal to truly embrace Kubernetes as the emerging container orchestration tool of choice—which allowed customers to run fleets of containers at scale and in unison—instead pushing forward with its own proprietary Docker Swarm orchestrator (RIP) with a myopic level of focus.

"The biggest mistake was to miss Kubernetes. We were in that collective thought bubble where internally we thought Kubernetes was way too complicated and Swarm would be much more successful," Jérôme Petazzoni, one of Docker's first and longest serving employees, said. “It was our collective failure to not realize that.”

The truth is, Docker had the chance to work closely with the Kubernetes team at Google in 2014 and potentially own the entire container ecosystem in the process. “We could have had Kubernetes be a first-class Docker project under the Docker banner on GitHub. In hindsight that was a major blunder given Swarm was so late to market,” Stinemates said.

Those early discussions at Google’s San Francisco offices were technical and tense, according to multiple people who were in the room, as both sides had strong opinions on how container orchestration should be done.

Craig McLuckie, Kubernetes cofounder and now vice president at VMware, says he offered to donate Kubernetes to Docker, but the two sides couldn’t come to an agreement. “There was a mutual element of hubris there, from them that we didn’t understand developer experience, but the reciprocal feeling was these young upstarts really don’t understand distributed systems management,” he told InfoWorld. Others say discussions were more informal and focused on joint development of container technology. Either way, the teams never saw eye to eye and ended up going their separate ways, with Google launching Kubernetes itself in the summer of 2014.

Hykes disputes that Google offered Docker ownership of the Kubernetes project, saying they had “the opportunity to be part of the ecosystem like everyone else.”

Hykes does acknowledge that there were tensions between the Docker and Google teams at the time. “There was a moment when egos prevailed. A lot of smart and experienced people at Google were blindsided by the complete outsiders at Docker," Hykes said. "We didn't work at Google, we didn't go to Stanford, we didn't have a PhD in computer science. Some people felt like it was theirs to do, so there was a battle of egos. The result of that was not a good collaboration between the Docker and Kubernetes teams, when it really made sense to collaborate."

"That fundamental ego on one side and tension on the other with [Kubernetes cofounders] Joe Beda, Brendan Burns, and Craig McLuckie—who had strong opinions about the need for a service level API and Docker technically had its own opinion about a single API from a simplicity standpoint—meant we couldn't agree," Stinemates said.

Hykes admits that Docker was under pressure at the time to find an orchestration solution for customers who wanted to scale their use of containers, but that it wasn't obvious at the time that Kubernetes would be that solution. "Kubernetes was so early and one of dozens and we didn't magically guess that it would dominate," Hykes said. "It wasn't even clear how committed to it Google was. I asked our engineers and architects what to do and they recommended we continue with Swarm," he said.

Even McLuckie admits that he "didn't know Kubernetes would become *Kubernetes*. It's easy to look back on history and call it a bad choice."

However it went down, Kubernetes ended up winning the container orchestration battle, and the rest becomes a great "Sliding Doors" moment for the software industry.

How to choose a low-code development platform

By

“Kubernetes came and stole all of that thunder,” Jay Lyman, an analyst at 451 Research, said. “It represented Google’s use of containers and so in developing and open sourcing that, in many ways that eclipsed the attention around Docker. [Docker] viewed Docker Swarm as their way to monetize their software. If they could go back they probably would have integrated more closely with Kubernetes from the get-go. They were too focused on going it alone.”

“One of my deepest regrets is we didn’t find a way to negotiate this,” McLuckie said. “Docker offered something experientially remarkable and what Kubernetes delivered, from an experiential perspective, was less remarkable.” Or, as Docker cofounder Sebastien Pahl noted: “Simplicity has not won. I love Kubernetes, but it is not for normal humans.”

Tensions at the top

Docker eventually hit the peak of its hype cycle in the wake of its 2015 mega Series D funding round of $95 million at a “unicorn” $1 billion valuation.

“That set expectations very high and exposed some of the fundamental difficulties we would have as a company,” Steinmates said. “I think Ben [Golub, the CEO] had a different idea for the company than Solomon did and it shouldn’t be a secret that the two didn’t see eye-to-eye. There was a lot of involvement from the board in trying to keep the founder happy and give the CEO enough leeway to make the company successful. If it were up to Solomon, we would have stuck on the community-oriented route to create virality. If it was up to Ben, we would have moved more strongly to the business side earlier. That tension caused us to do both in a half-assed way.”

This approach effectively spawned two Dockers: Docker Community Edition, the massively popular command-line tool and open source project aimed at developers, and Docker Enterprise Edition, a suite of commercial tools aimed at enterprise customers who wanted to adopt containers at scale. Unfortunately, the company was too slow to make that split official and divide resources accordingly.

Golub admits that they “should have split the business” earlier than they eventually did, while Hykes agrees that Docker “never found a way to connect those two halves of the company.”

By 2018, the cracks were starting to show, as the company struggled to steer a viable course between an increasingly disgruntled open source community, powerful partners, and demanding enterprise customers trying to run containers in production.

Hykes left his day-to-day role at the company soon afterward, in March 2018, noting in a blog post that, “As a founder, of course, I have mixed emotions. When you create a company, your job is to make sure it can one day succeed without you. Then eventually that one day comes and the celebration can be bittersweet. It’s never easy for a founder to part ways with their life’s work.”

Looking back today, Hykes is more succinct. “I realized that I didn’t belong at the company. It wasn’t constructive for me to stay, so I left … I was mostly an unhappy founder who should have stayed on as CEO or left.”

Docker breaks in two

In the face of growing funding troubles, Docker cycled through new CEOs, with Golub giving way to former SAP executive Steve Singh in

May of 2017, and then Singh stepping aside for former Hortonworks CEO Rob Bearden in June 2019.

It fell to Bearden to finally face the music. Shortly after he took the helm, Docker sold the enterprise part of its business to Mirantis in November 2019, and Docker Enterprise was absorbed into the Mirantis Kubernetes Engine.

“After conducting thorough analysis with the management team and the board of directors, we determined that Docker had two very distinct and different businesses: one an active developer business, and the other a growing enterprise business. We also found that the product and the financial models were vastly different,” Bearden said in a press release at the time.

Where is Docker today?

The remains of Docker, anchored by the original Docker Engine container runtime, Docker Hub image repository, and Docker Desktop application, still lives on today under the leadership of seven-year company veteran Scott Johnston, thanks to a $35 million cash injection from original investors Insight Venture Partners and Benchmark Capital.

Johnston is attempting to take the company back to its roots, by focusing “laser-like back on the needs of developers,” he told InfoWorld. “We think the company is in a stronger than ever position because of three things: customer focus, aligned go-to-market, and an ecosystem-friendly business model.”

Last week Docker announced changes to the Docker software’s licensing terms. Soon professional users of Docker Desktop who work for large companies will have to sign up for a paid subscription to continue using the application.

Johnston is determined not to repeat the mistakes of the past, by focusing on delivering value to the company's core audience of software developers. "Our ambition is greater, as the addressable developer ecosystem is every developer in the world, not just those aligned to our runtime," he said.

Where Johnston sees growth opportunities for "Docker 2.0" is in building new developer tools and trusted content for secure, verified images, as well as continued momentum behind emerging computing models like serverless, machine learning, and IoT workloads, which are underpinned by container technology.

Meanwhile, Docker remains the industry standard container runtime, with Docker Desktop installed on 3.3 million machines today. Further, 49% of respondents to Stack Overflow's 2021 developer survey said that they use the tool regularly.

Nonetheless, there remains a deep sense of disappointment around what could have been. "If I wanted to be flippant, I would ask if Docker even exists today," Stinemates said. "From a career perspective, it's sad. I'm still looking for a company as exciting and dynamic and to create a spark like Docker did."

"It is fair to say Docker failed to realize its potential as a business ... so far. I am very excited that Docker gets another shot at building a business after so many years. That is testament to the underlying project and the brand," Hykes said.

Scott Carey is the Group Editor for IDG UK enterprise titles, writing

primarily for InfoWorld.

How to choose a low-code development platform