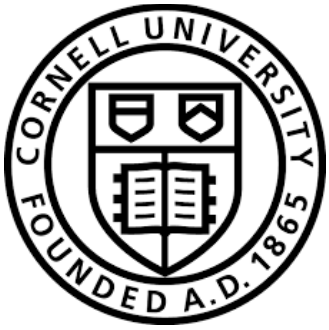
Public Sector | Customer Narrative

Cornell University Simplifies Production Deployments with Docker Datacenter



Direct line of support to the Docker Customer Success team to quickly report and remediate any issues with their environment.

BACKGROUND

Located in Ithaca, New York, Cornell University is an ivy league educational institution that was founded April 27, 1865. The faculty at Cornell is engaged in many projects in the sciences, engineering and more. Cornell's online library, ArXiv.org, serves 30,000 students and faculty and receives close to 1.5 million hits per day from all the world, with 14 million papers downloaded per month. Cornell's priority was to modernize this important educational and research tool.

CHALLENGE

The ArXiv online library is core to the educational experience at Cornell and more important enables many research projects that provide external funding to the University. Research is an important factor in maintaining the reputation of Cornell University as an innovative environment for students and faculty.

What needed to happen with ArXiv:

- ArXiv is constantly evolving and the team was looking to modernize this aging application so the frequent enhancements could be done faster.
- By simplifying maintenance and management of ArXiv, Cornell could lower the cost of managing and redirect dollars to critical enhancements.

THE JOURNEY

Cornell had previous experience with Docker Datacenter. Confluence was the first production application Cornell moved to Docker. Prior to moving Confluence to Docker and to the cloud, Cornell was spending 1770 staff hours supporting Confluence. After the move to Docker this was reduced to 178 hours – a factor of 10x improvement that enabled Cornell to redirect resources to other important projects like ArXiv. Docker Datacenter provided Confluence with reproducible infrastructure. After adopting Docker Datacenter the base images are rebuilt with the latest patches, on a daily basis.

ArXiv was a monolithic legacy application. Required enhancements were taking too long to develop and move to production. Transforming the ArXiv application to microservices would accelerate time to new value. Like Confluence, ArXiv was old code that had been maintained manually. Would Docker Datacenter's ability to provide infrastructure as code simplify the challenges of managing microservices? Cornell wondered if moving ArXiv to Docker Datacenter would provide the same improvements for ArXiv as it did for Confluence?

"I urge you to consider Docker when looking at your legacy and vendor applications. You will be surprised by the efficiencies you find. Also, do yourself a favor and look at Docker Datacenter. We have benefited greatly from Docker's commercial support and the relationships we have made within Docker the company."

– Shawn Bower, Cloud Architect
Cornell University

SOLUTION

Empowered by their Docker experience with the Confluence application, Cornell implemented the move of ArXiv to Docker Datacenter. The positive results provide many quantifiable benefits for Cornell:

- Portability of applications across the application lifecycle. Cornell benefits from the streamlined workflow and the ability to track changes.
- A central location for hosting Docker images and enables multiple organizations secure access to the images. This increases productivity for developers.
- High availability set up with secure registry replicas ensure continuous availability. The secure image storage allows the University to comply with industry standards and reduces risk.

One major benefit of Docker Datacenter is the speed it provides to build and ship applications – new applications and in this case a legacy application. Cornell is deploying applications 13 times faster by leveraging the reusable architecture patterns of Docker Datacenter and simplified build and deployment processes. Equally important is the manageability of microservices. Docker Datacenter provides a platform that is resilient, reliable and available.

ABOUT DOCKER

Docker is the leading software container platform. Developers use Docker to eliminate "works on my machine" problems when collaborating on code with co-workers. Operators use Docker to run and manage apps side-by-side in isolated containers to get better compute density. Organizations use Docker to build agile software delivery pipelines to ship new features faster, more securely and with confidence. www.docker.com/government

www.docker.com

docker