

Faire Revamped Kubernetes-based End-to-end Testing Environments with Crafting

“Crafting has been a step change in our developer experience at Faire.”

Executive Summary

Who:

Faire is a wholesale marketplace that connects independent artisans and brands with local retailers. The platform connects independent retailers, across 50,000 cities, with 100,000 brands from around the world. This huge global platform is supported by a strong product & engineering team with several hundred engineers for innovations and scaling.

Challenges:

- Scaling the product and building out microservices
- Docker-compose does not scale for a growing number of services
- On-demand production-like environments are critical for end-to-end testing
- Most engineers don't have the desire to learn how to operate Kubernetes and Terraform
- Cost could be high for maintaining many end-to-end environments

Solution

- Let Crafting orchestrate end-to-end testing environments in a testing Kubernetes cluster
- CI-bot uses Crafting to launch end-to-end testing environments for every needed PullRequests (PRs) automatically, reusing existing config for production deployment
- Allow engineers to keep an environment to quickly “debug in-place” for failed tests
- Let engineers to connect frontend to a sandbox environment to test and iterate quickly

Results:

- ✓ Approximately 1000 sandboxes created daily on Kubernetes for end-to-end testing
- ✓ Step change in developer experiences for end-to-end testing at Faire
- ✓ It encourages engineers to test more and identify issues earlier, thanks to the easy-to-use and scalable end-to-end environments with high fidelity to production
- ✓ Built-in monitoring and auto-recycling mechanism alleviates management overhead while keeping cost low

Faire operates as a wholesale marketplace facilitating connections between independent artisans and brands with local retailers. The platform boasts an extensive network of retailers spanning 50,000 cities, fostering 7 million connections globally with over 100,000 brands. This expansive global platform is underpinned by a robust product and engineering team comprising several hundred engineers dedicated to driving innovation and scalability.

The Challenges

To sustain its rapidly expanding global business network, Faire has a multitude of backend services and continues to add more. The conventional method of running these services locally via docker-compose gradually became less and less viable due to resource limitations and the complexity of diagnosing issues. At the same time, a different approach was used in the Continuous Integration (CI) process for end-to-end testing, creating disparities between the staging and production environments. Maintaining three disparate deployment methods for the same set of services not only imposed a substantial maintenance overhead but also resulted in a suboptimal engineering experience, as exemplified by the common challenge of "working locally but failing in the production."

Faire recognized the need to address the difficulties posed by the local docker-compose setup and opted for a standardized method of deploying services on Kubernetes for development and testing. Given Kubernetes is adopted in the staging and production environments, this approach would also enhance observability and a closer alignment between the development/testing systems and the staging/production ones. Clearly, a system built around on-demand environments that mimic production is necessary for efficient end-to-end testing.

“

It became clear that eliminating the challenges of the local docker-compose setup in favour of a standardized way of deploying our services into Kubernetes for dev/test purposes would be ideal.

However, to adopt such a Kubernetes-based approach, a few challenges need to be addressed. First, most engineers who need to work with the testing system on a daily basis lack the specialized DevOps expertise required for Kubernetes and Terraform. They have little motivation to acquire these skills due to the steep learning curve and the lack of relevance to their primary responsibilities. Combined with the necessity for safeguards to prevent a single engineer's misoperation from disrupting the entire testing system, there is a pressing need for an intuitive and user-friendly platform built on top of Kubernetes to cater to end-users.

In addition, there is the potential concern of resource costs associated with managing numerous copies of end-to-end (e2e) testing environments. While an individual e2e testing environment may not require substantial resources for running tests, the cumulative resource expenses can

escalate rapidly when maintaining hundreds of such environments to accommodate every engineer's needs. Coupled with the tendency of engineers to forget manually releasing resources, the automatic lifecycle management becomes an imperative need for the system.

The Solution



We considered building a platform to manage these sandbox environments ourself, but also looked at third party options, and eventually settled on crafting.dev as our preferred solution.

After careful consideration of multiple options including building internally and using third party systems, the Faire team chose Crafting as the preferred solution to integrate with. Specifically, the platform engineering team in Faire leverages two key features Crafting offers: a) lifecycle management of end-to-end testing sandboxes on the testing Kubernetes cluster, and b) intuitive and customizable UI for great developer experience.

Using the Crafting system, the Faire team created a testing template that encapsulates the end-to-end (e2e) environment running numerous services within a dedicated Kubernetes namespace. Engineers can then conveniently launch these environments on demand. Each PullRequest (PR) with a tag indicating the need for e2e testing will trigger the git hook to create such an environment and run corresponding e2e tests on it. When the tests finish, the environment can be recycled quickly and automatically, freeing up the resources promptly without additional intervention from engineers.

An essential aspect of this approach is the reuse of config files for launching the services in e2e Kubernetes namespaces, drawing from the production config. Compared to creating a new set of custom configurations or relying on docker compose, this approach not only reduces the onboarding efforts, but also simplifies the ongoing maintenance, preventing divergence in multiple environments. The capability to test in a truly production-like end-to-end environment significantly increases overall reliability by reducing the “surprises” when code is deployed to production.

Another key advantage of this approach lies in the capability to let engineers view logs easily and debug in place. It's a long-standing pain point for engineers to debug issues found in integration testing, which can't be done with good observability into the environment. At Faire, the on-demand Kubernetes environments are not only hooked with great logging support, but can also be retained in their current state for engineers to debug in case of test failures. Crafting also provides tooling to let engineers get into pods in the connected testing Kubernetes cluster, so that they can inspect the exact environments to validate assumptions, which is very useful to catch configuration issues.

Last but not least, Faire engineers can connect their frontend to an end-to-end sandbox for their frontend development. Instead of waiting for the corresponding backend changes to be merged and deployed to staging or production, the engineers at Faire can develop the frontend by directing their API to an end-to-end sandbox. This has significantly accelerated the iteration process for modifications that affect both the frontend and backend, leading to more efficient development cycles.

The Results

Faire solved the issue of standardizing the dev/tests environments and launching on-demand end-to-end testing namespaces with a solution based on Crafting. Now the system is running at scale with around 1000 daily launches of such end-to-end sandboxes, verifying code changes from the engineering team with hundreds of engineers.



Crafting has been a step change in our developer experience at Faire.

The great user experience is cheered by engineers. When the platform engineering team presented the new end-to-end sandbox solution in an internal tech talk, the engineers were very excited about the solution and shared their excitement in channel messages. Within one minute, the message board was filled with comments like “Wow”, “This can’t be real! Too good to be true”, “OMG this is awesome”, “I’ve never run BE locally. And now I never will”. Powerful and convenient tooling encouraged engineers to test more and catch issues earlier in the development cycle.

On the other hand, this powerful solution at Faire also keeps resource costs low. Thanks to the automatic lifecycle management, sandboxes are recycled quickly and the corresponding Kubernetes namespaces are cleared promptly when the tests finish, unless there are some test failures requiring engineers to look into the environment. This efficient process enables a high volume of tests to run through the system without an outsized node pool.

Furthermore, Faire minimizes the burden of maintaining such a system by leveraging Crafting’s managed self-hosting solution. In this setup, the Crafting system is hosted on the customer side but is continuously monitored and managed by the Crafting team, which includes handling all updates and configuration changes to fit customers' needs. “Crafting's managed self-hosting solution keeps our maintenance burden low and they are very responsive to both feature requests and to help with any troubleshooting needed.”, commented by Ben Poland, Staff Platform Engineer at Faire, “The integration process was very smooth and flexible, allowing us to adapt our existing tools and processes to get things working faster than I expected.”

Next Step

Going forward, Faire is looking into advanced Kubernetes debugging features such as Traffic Interception to virtually replace services in the end-to-end namespaces with a locally running version, which significantly accelerates iteration and could make debugging even easier.

In addition, Faire plans to investigate the remote development opportunities that the Crafting platform also provides in order to accelerate our development teams even more.

Interested?

Please contact us at contact@crafting.dev to learn more or see a demo.

