

## THE CUSTOMER

[Mettle](#) is the digital business account by Natwest that provides free accounts for startups, small businesses, limited companies and sole traders. Launched in 2018, Mettle is part of NatWest's business banking innovation arm. It operates like a tech startup with its own office space in London.

Mettle's goal is to provide small business customers with an application that speeds up payments and helps them manage their finances. One third\* of small businesses suffer problems due to late payments and lack of visibility, which is why the Natwest team have set out to revolutionise business banking. Customers can apply for an account in minutes, create and send customized invoices and connect to accounting software to help keep track of their books.

## CHALLENGES

To keep the Mettle application running, and ensure that all services are available to their customers 24x7, the engineering team needed a way to deploy new features quickly without compromising on reliability or security.



*One of my team's biggest principles is to provide a self-service platform for our engineers. GitOps is making this possible."*

- Steve Wade, Platform Lead

## Mitigate the risk of compromised CICD pipelines

One of the main changes Mettle's platform team wanted to implement was ensuring that the engineer's CI tooling (concourse) was secure and wasn't an attack vector into other environments. If hackers were to penetrate a cluster, the first place they would go to would be the CI machine(s) because they always have elevated privileges to access environments/clusters. Prior to implementing GitOps, Concourse had the ability to create, edit and delete almost anything in all of their clusters. This meant that the current state of a cluster could only truly be determined by Concourse itself.

# mettle.

**Industry:** Finance

**Location:** London

## HIGHLIGHTS

- Production speed increased by 50%
- Deployments increased by 75%
- MTTR in 20min for all clusters
- Devs 75% less focused on operations

## KEY BENEFITS

- GitOps enabled a self-service Development Team
- Weave Flux enabled greater security into their workflows

## CONTACT US



[www.weave.works](http://www.weave.works)



[sales@weave.works](mailto:sales@weave.works)

## Decrease operational overhead

Another challenge was that the Platform Team, who is responsible for the creation and deployment of all infrastructure and Kubernetes clusters, needed a self-service platform. This would empower their developers to innovate and deploy microservices all the way to production with no input required from the team. Furthermore, with Mettle being a Fintech company, they are regularly audited for compliance, so they needed a way to show auditing logs.

## SOLUTION

Mettle's engineers were already using Concourse CI, which they had invested a lot of time and effort into. They chose Weave Flux because the Platform Team wanted to leverage a tool that would integrate well with the engineer's existing pipelines. Mettle wanted the focus of their engineers to be on building assets, testing them in specific environments and allowing an agent to handle how those assets get deployed. Another key reason for choosing Weave Flux was from a



*Leveraging GitOps has allowed us to create a self-service platform for engineers so they can concentrate on delivering business value through innovation, without the full need for Platform Team assistance. The engineers focus on building container images and managing the testing of their microservices and Weave Flux handles the deployments.”*

security perspective. Having something pull changes into the cluster rather than push them in from an external system meant that Mettle only had to handle the creation and revocation of GitHub deploy keys, which they do via Terraform. These keys are used by their Weave Flux instances to communicate with the necessary repositories.

## GitOps workflows increased productivity

Mettle has multiple Weave Flux instances running on their clusters, reconciling resources from Sealed Secrets to Kafka Topics. They prefer to have multiple repositories for different resources because they require different owners and teams to approve Pull Requests.

GitOps was introduced when Mettle was migrating their existing infrastructure to a re-designed, new platform called “v2”. For the migration they had a core team of engineers who were trained on Weave Flux, GitOps, Kustomize and Helm.

That knowledge was then spread amongst the wider engineering team, via documentation, pull requests and further collaboration.

## Gain full compliance with built-in audit trails

Before leveraging GitOps, the engineering team deployed rogue or inaccurate manifests to the clusters due to a lack of linting in their Concourse pipelines. Now thanks to GitOps, all cluster changes are driven via Pull Requests and changes to key components must be reviewed by certain teams. Engineers are now enabled to deploy microservices all the way to production with no input required from the Platform Team.

## RESULTS

### Reproducible, consistent clusters

GitOps allows Mettle to have truly ephemeral clusters. They can now always reconcile the current state of any of their clusters from multiple Git repositories. An example of this was a requirement for a separate environment to serve a Bug Bounty program. This environment needed to closely mirror their production environment. Their HelmReleases for production leveraged the Weave Flux annotation `weave.fluxcd.io/application.tag: glob:prd-*``. They used the same annotation for their Bug Bounty HelmReleases to ensure that the exact same microservice versions are maintained in both environments with zero changes required to their CI pipelines. The results are that Mettle can create these environments in under half a day using both Terraform and Weave Flux.

### Self service cluster management

Mettle increased their deployments to production by 50% by implementing a self-service platform which better served their development team. The team is now more confident in the overall stability and auditability of what they have built. They are able to destroy an entire cluster and bring a new one back to life in around 20 minutes with full confidence that it will be in the same state as before. This positive change in their Mean Time to Recovery (MTTR) is across all of their Kubernetes clusters. Their most recent number of Production releases was 218.