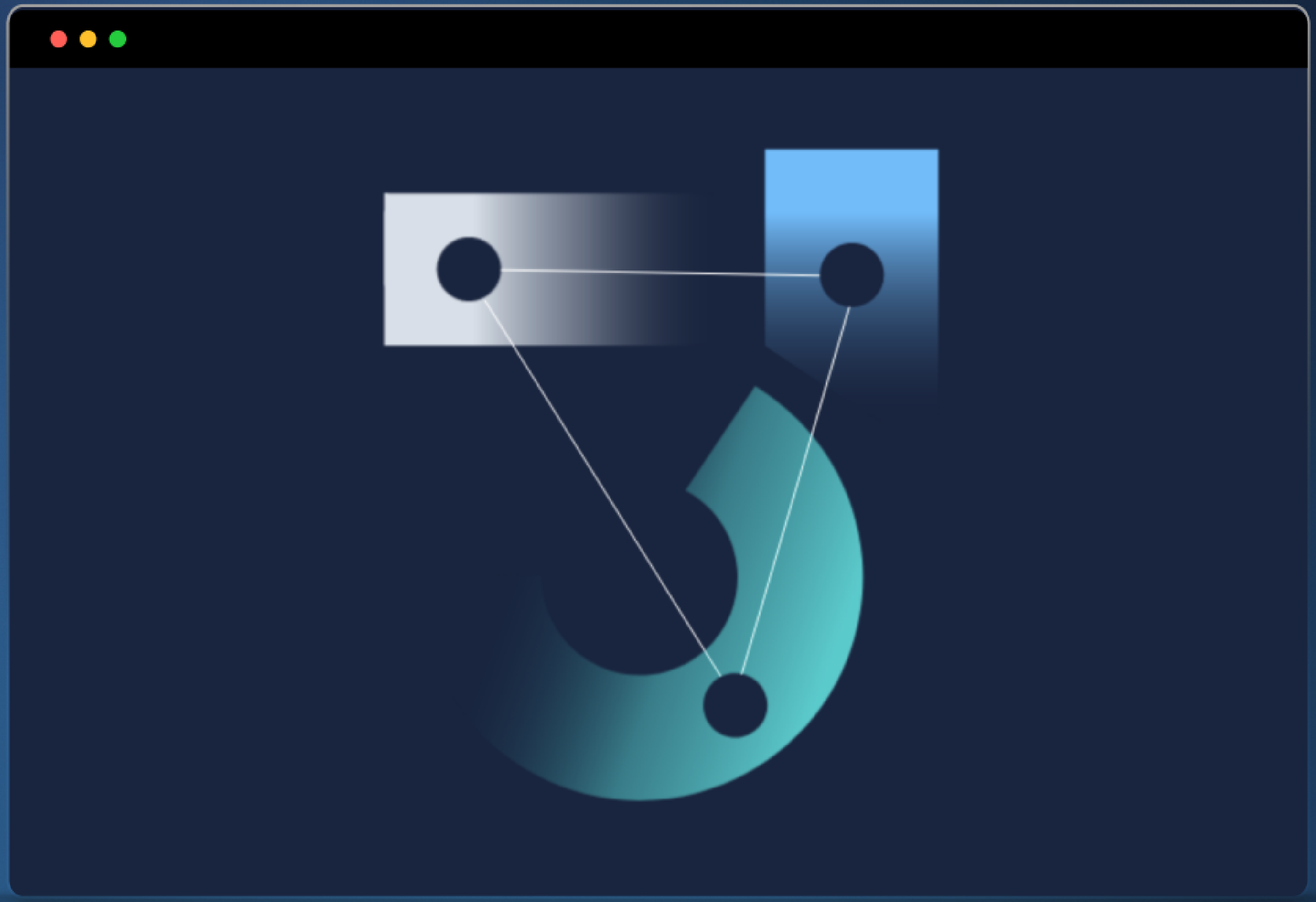
Cockroach Labs

[CASE STUDY]

FORM3

Formed on a multi-cloud platform: scalable, resilient payment technology



FORM3

How CockroachDB helps Form3 mitigate risk and reduce payment complexity for their customers

INDUSTRY:
Fintech

CHALLENGES:
Building a cloud-native, enterprise-grade payment technology using microservices and Kubernetes.

SOLUTION:
A highly scalable, adaptable, resilient, multi-cloud platform built on CockroachDB.

About FORM3

When you send money to another person who banks with a different financial institution, the banks talk to one another to confirm each account holder is legit. Then they physically move the money. Sounds simple, right?

This straightforward process gets complicated, though, when you serve a global customer base and need to navigate different payment schemes (i.e., the set of rules for performing payment transactions) in different countries. Form3 was founded to make this much less complex for its customers by delivering a managed payments infrastructure with a single API that helps you adhere to payment schemes wherever you do business.

Building a cloud-native, enterprise-grade payment technology using microservices and Kubernetes is no easy undertaking. Especially when that platform has to be highly scalable, adaptable, and resilient — which is why Form3 turned to CockroachDB.

Get in formation

Based in London, Form3 started its business to cater to banks and fintechs that do business in the UK and EU. Within these regions, there are five different payment schemes you must integrate into to transfer funds. While they all have the same function — transferring money — the implementation, requirements, and payment schemes are all different for each one. What's more, these schemes are constantly changing.

For example, when connecting to the UK's real-time Faster Payment System (FPS), companies are required to have leased lines, physical HSMs, and other hardware, which means they cannot operate exclusively in the cloud. This is a challenge especially for digital-first mobile banks that want to get their MVP to market quickly by leveraging the convenience of a public cloud provider. However, to integrate to FPS, they would be required to have their own equipment, connect to physical data centers, and continually manage the integration's frequent changes.

Beyond different schemes to navigate, there are also different legislations Form3 must adhere to. Several large financial institutions are well on their way to becoming cloud-first and regulators are nervous about these companies all choosing the same cloud vendor. In the UK, there is growing concern about what would happen if all banks were running on the same public cloud provider and some type of incident occurred (e.g., a major outage, sudden steep price hikes, etc.). One public cloud provider's unfortunate event could have dire consequences for the entire UK economy.

Form3 originally selected AWS as their cloud provider and they started building their FPS access solution on Amazon RDS for PostgresSQL, which initially worked well for their needs. They liked that PostgreSQL was reliable and could deliver data consistency for payments.

> *"It's the best of NoSQL in terms of horizontal scale, and the best of relational in terms of ACID compliance and write consistency so we don't lose any payments. It's the only offering we saw on the market that really solves that problem. And coupled with the fact that it's super easy to run in Kubernetes, CockroachDB became an easy choice."*
>
> – **Kevin Holditch**
>
> Head of Platform Engineering, FORM3

However, regulators started to ask Form3 questions about their infrastructure: What would happen if you couldn't use AWS anymore? What's your cloud strategy? Is your platform portable? What if AWS has an outage? Can you run your platform in different clouds?

Given all the emerging regulations, they thought there was a better solution: **to not depend on any single cloud provider and run across multiple clouds at the same time**. Once they landed on this approach, these were the requirements for a new database:

- ✓ Cloud-neutral, supports multi-cloud
- ✓ SQL compatibility
- ✓ Easy to run and maintain
- ✓ Strongly consistent
- ✓ Scalable
- ✓ Easy to run with Kubernetes

Distributed databases like Aurora and Spanner met a lot of these requirements, but they are both tied to a particular platform vendor and so are not cloud neutral. Other databases that provided the necessary scale and performance (like Cassandra and Redis) didn't prioritize data consistency. This set of requirements ultimately led them to CockroachDB.

CockroachDB fulfilled the consistent storage requirements and can be run across Kubernetes clusters. Not many database solutions have the same distributed nature as Kubernetes, which can make deployment and management difficult.

3

PUBLIC CLOUD PROVIDERS

24/7

AVAILABILITY

5

PAYMENT SYSTEM INTEGRATIONS

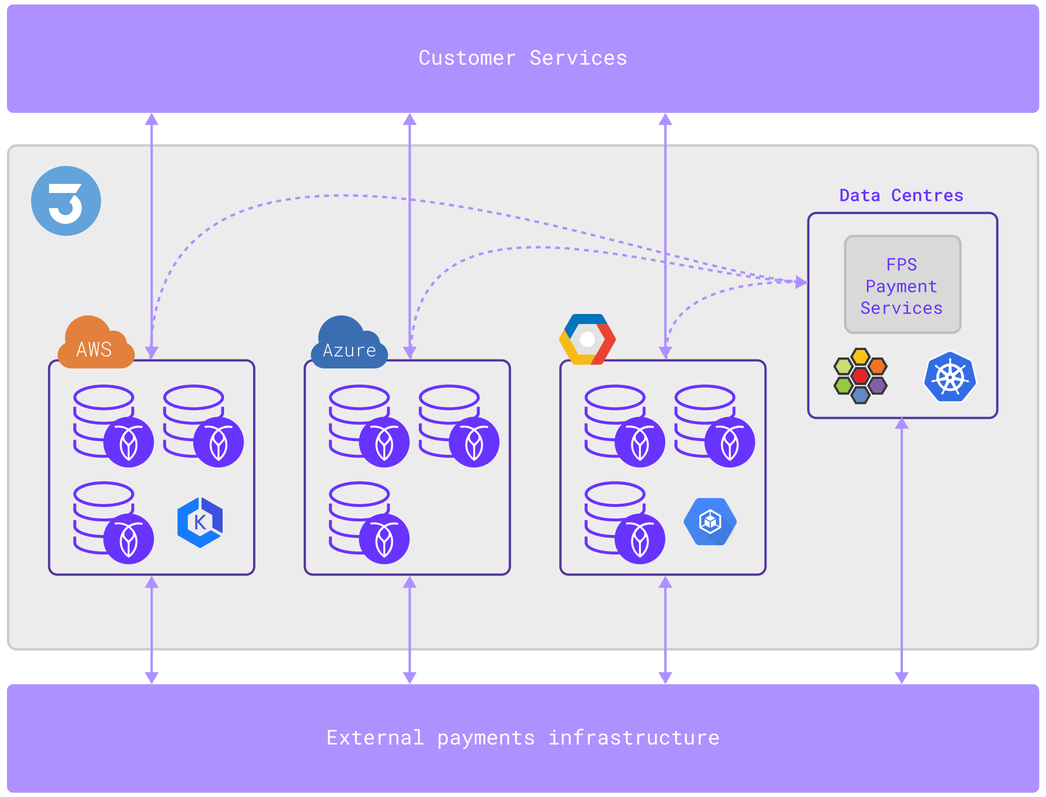
Form follows function

Even though Form3 wanted to run completely in the cloud, they knew that they had to have a combination of their own data centers and public clouds in order to connect with certain payment schemes. After choosing to migrate their FPS access solution to CockroachDB, they began to think about a different deployment methodology for their new platform that would satisfy regulator requests. They needed a second cloud.

They thought they could have replicated the entire platform onto GCP, which has a matching set of technologies to AWS (i.e., instead of SQS they use PubSub). The problem with this approach is that they would be managing two versions of their platform that have different behaviors. This was fairly tough to do and would create a lot of maintenance overtime.

Instead, they decided to run the whole database across all three cloud providers: AWS, GCP, and Azure. They use the managed Kubernetes control plane offering for each vendor so that the vendor runs Kubernetes. That way, the Form3 engineers can concentrate on the value add for the business, which is running their software and configuring CockroachDB on top of it.

For their setup, their data has a replication factor of three, and each range is replicated across each cloud and they are able to maintain strong consistency on each range. Here's what it looks like:



With CockroachDB, you can configure different topologies based on where your customers live. So for example, if you have a customer in London, you set it up so CockroachDB fetches data from the closest node to their London location and you get great performance. If that region goes down, though, there's still a replicated copy of that data in a different location and on a different cloud provider.

Form3's multi-cloud deployment is advanced. They had to privately network all the clouds together and maintain network connectivity through multiple datacenters. Networking across clusters is not simple, but CockroachDB's operational simplicity can make it easier. By leveraging VPC peering and assigning Kubernetes pods their own IP address, Form3 can point CockroachDB nodes at one another and the pods will end up talking to each other from different clusters.

When you are federating Kubernetes clusters, if a pod goes down it typically takes other pods down with it. However, CockroachDB allows you to federate pods at the data layer. This means the clusters are running independently and so don't really "know" about each other. Because they are isolated in this way, they won't take each other down. This helps mitigate a great deal of risk.

> *"CockroachDB is doing some amazing gymnastics under the bonnet. You might think it's just a simple query, but it's actually going off to different nodes where the data is physically stored to retrieve it. The performance is really good. And then you have this tremendous scaling capability. Using CockroachDB almost feels a bit magic."*
>
> – **Kevin Holditch**
>
> Head of Platform Engineering, FORM3

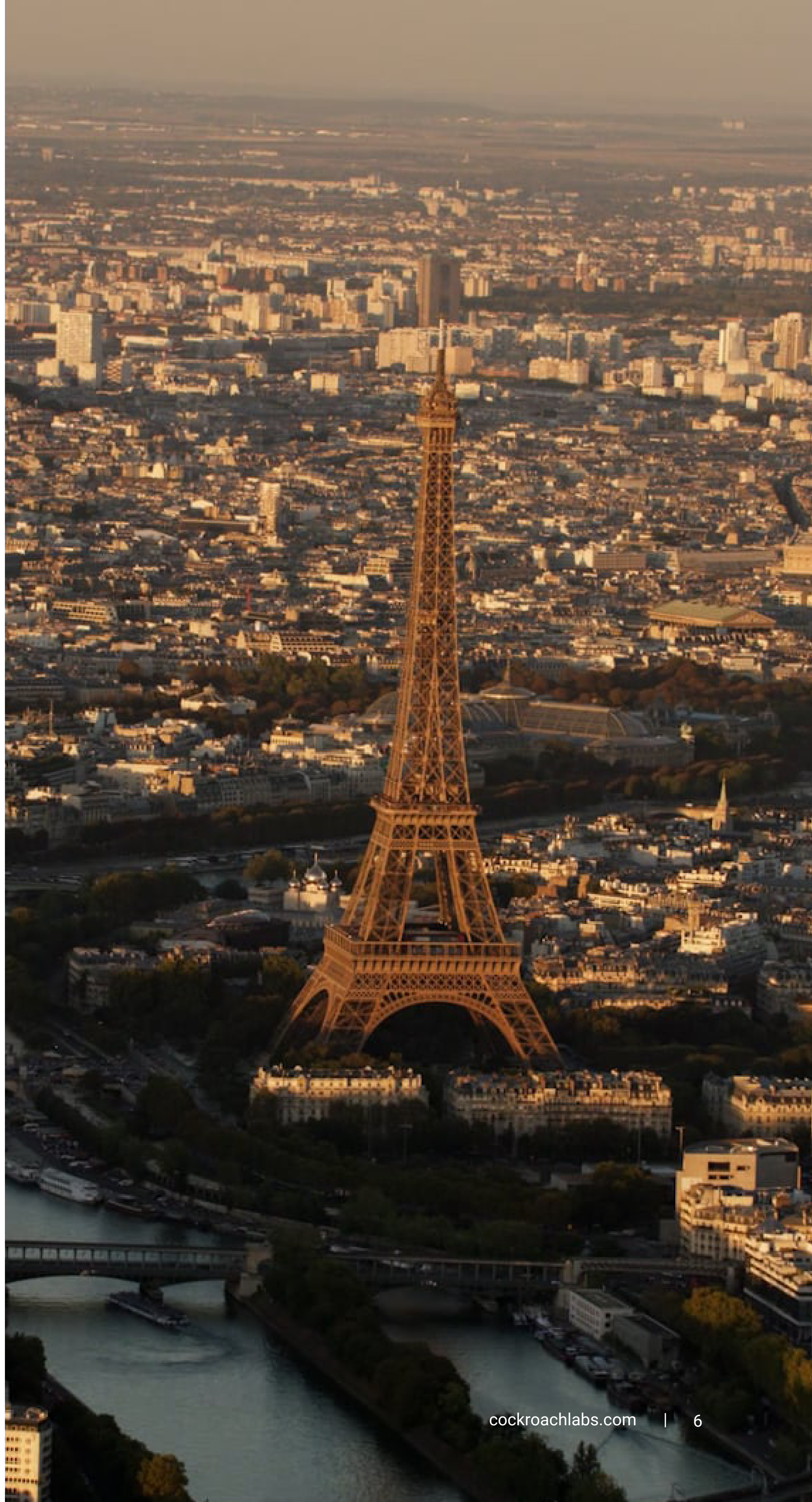
Perfect your form

When it came to initially migrating the FPS access solution to CockroachDB, it was a “fairly easy process” for the Form3 team. Since CockroachDB is PostgreSQL wire-compatible, Form3’s data structures and queries just worked — with a few that they had to tweak. Most of the Form3 platform is written in Go. A majority of existing PostgreSQL tools also work with CockroachDB, so they could leverage the Go drivers they were already familiar with.

The Form3 team recommends setting up a good test suite and running end-to-end tests that include the database. They would even drop the database connection to ensure that the software handles the failure correctly. They also tested latency to make sure that their payment journeys were rock solid before completing the migration.

CockroachDB also has a Kubernetes Helm chart available, packaged up nicely for an easy install. And, as mentioned above, Form3 uses each cloud provider’s managed Kubernetes offering that alleviates a lot of operations for their team.

The Form3 team highly recommends looking into managed Kubernetes; otherwise, you might end up needing an army of experts to run and maintain the platform. There’s not a competitive advantage to managing yourself. If Kubernetes goes down, Form3 has enterprise support with the cloud vendor so they can get someone to help. And, since they are using CockroachDB, they are not locked into a single provider, which does give them a competitive edge.



What's Next

Form3 is on a mission to become the world's most trusted provider of payment technology. They remove the burden of managing costly, evolving, critical payment infrastructure from their customers, which allows them to seamlessly add payment schemes as they expand into new regions. With CockroachDB as a crucial part of their platform's foundation, Form3 can navigate emerging legislations and continue to grow their business.

Today, Form3's API-based cloud technology solutions are used by financial institutions such as Goldman Sachs, Lloyds Bank, and Nationwide Building Society to deliver mission-critical payments. While a majority of their customers are based in the UK and the EU, they are expanding to the US and other international markets.

Going cloud-native across clouds with Form3 ▶

Ready to get started?

Go hands-on with 100% free CockroachDB Serverless. Spin up your first cluster in just a few clicks.

Get Started →   Compare Products →