

# CASE STUDY

improving 



## BERKSHIRE HATHAWAY ENERGY (BHE)

### PROJECT

DekaFlow

### OVERVIEW

Berkshire Hathaway Energy (BHE) is a oil and gas conglomerate managing assets across the entire O&G vertical. They recently acquired Dominion Energy for their pipelines and processing facilities. With the Dominion Energy purchase came the management and nominations software for those new assets. The existing software was outdated and subject to unfavorable licensing terms, and difficult to integrate with the existing BHE technologies. BHE approached Improving to modernize their legacy Java Swing app and move it to a cloud- native, React, and Java microservices architecture. The client's business problem was to decouple themselves from Oracle's licensing agreements and move into cloud-native technologies to increase uptime, reliability, and decrease downtime in the event of failure. The solution was to take their existing application and write it as a cloud-native microservice stack with modern technologies deployed in an Azure environment.

### BUSINESS PROBLEM OR CHALLENGE

BHE had legacy Java Swing desktop apps with on-premises servers for all their back-end data in Oracle databases. They were trying to decouple themselves from Oracle's licensing agreements and move into cloud-native technologies to increase uptime, reliability, and decrease downtime in the event of failure with regional and multiregion failovers. The challenge was to modernize their application and move it to a cloud-native, React, and Java microservices architecture.



# CASE STUDY



## SOLUTION / OUR APPROACH

The Improving team analyzed the legacy Java Swing app and migrated functionality to new microservices while maintaining data integration with the legacy systems. They also provided guidance on systems architectural design. The team used Kafka as the backbone messaging fabric, which helped move them towards microservices, event sourcing, and domain-driven design while consuming and providing data to the legacy systems. They also wrote first-class front-end websites using React and Websockets, allowing the front-end to interact with Kafka securely and efficiently.

## BUSINESS BENEFITS

BHE saw a significant improvement in their feature development life cycle. It was a lot faster to develop new features and iterate on existing ones. They were deploying two or three features per Sprint, whereas before it was much slower with release cycles and release management workflows to validate and release new features. The team successfully deployed 5 microservices into Azure serving the client's multiple pipelines and a processing facilities. The client signed on with Kafka and Mongo as service provider partners, so they have support for that. The resources in Azure are developed and deployed via Terraform so they can spin up new environments as they need to. In case of failure, they can spin up a new environment within like 5 minutes.

## QUANTITATIVE RESULTS

The client saw a significant improvement in their feature development life cycle, which was rapidly improved. It was a lot faster to develop feature and deploy features.

## QUALITATIVE BENEFITS

The client experienced a lot of qualitative benefits, including a significant improvement in their feature development life cycle, which was rapidly improved. It was a lot faster to develop feature and deploy features.

# CASE STUDY



## TECHNOLOGIES USED

The Improving team used a variety of technologies to modernize the client's application, including Kafka, React, Java, Terraform, and MongoDB, and MongoDB Atlas for full-text search.

## PARTNERSHIPS USED

The Improving team partnered with Mongo DB to provide their database needs for this project because they were doing a lot of that domain-driven design microservices. Where document databases work better and additionally partnered with Confluent to provide Kafka.

## CONCLUSION

In conclusion, Improving successfully modernized BHE's legacy Java Swing app and moved it to a cloud-native, React, and Java microservices architecture. The client saw a significant improvement in their feature development life cycle, which was rapidly improved. They were deploying two or three features of Sprint, whereas before that was much slower release cycles and release management to create release features. The team successfully deployed 5 microservices into Azure across the client's three pipelines and a processing facility that moves. The client signed on with Kafka and Mongo as service provider partners, so they have support for that. The deployables in Azure are developed and deployed the Azure infrastructure using Terraform so they can spin up new environments as they need to. In case of failure, they can spin up a new environment within like 5 minutes.

## GET STARTED

Learn more about how Improving can help you get started by contacting us today at [sales@improving.com](mailto:sales@improving.com).