

# Built to Last. Tchibo's Webshop Evolution into the Cloud

How freiheit.com engineered a seamless transition during live operations



x



# CONTENTS

1

From on-premise to infinite scale

2

Meet the key players

3

Defining the problem

**4** Milestones

**5** Looking back

**6** Get in touch

# From on-premise to infinite scale

Tchibo is the biggest coffee retailer in Europe, with more than 9 Million webshop visitors each month and a turnover of 3 Billion Euros.

Since 1949 they ship roasted coffee beans to their customers. Nowadays they have updated their assortment with a wide variety of non-food products and are known for their weekly changing retail goods – their “Themenwelten – Jede Woche eine neue Welt” (eng. theme world – each week a new world).

Tchibo truly was an internet pioneer and launched their first webshop in 1997. From then on their customers could order coffee as well as the whole non-food assortment online.

In the beginning Tchibo used a standardized webshop solution. But this system couldn't keep up with the demands of Tchibo's continuously growing assortment and reliably launching “Themenwelten – Jede Woche eine neue Welt”.

In 2008 they found a reliable partner to build a customized new webshop tailored to their needs – the software engineering company freiheit.com. Building this new system from scratch, enabled Tchibo to efficiently and productively manage and maintain their increasing webshop assortment.

The new webshop was seamlessly scalable, ensuring smooth operations even on the busiest days – especially during peak holiday seasons. This not only improved Tchibo's webshop operations but also their internal processes.

Read how freiheit.com did that [here](#).

For over 16 years, freiheit.com made sure Tchibo's operations were efficient and ensured a unique shopping experience for their customers even with upcoming demands.

When you build and maintain such a system over almost two decades at scale you learn a lot how to build systems that can stand the test of time. freiheit.com added a lot of new features, rolled out to new countries and the traffic quadrupled over time. And still it is a solid system that Tchibo can trust.

But one thing was certain – It also needed to remain a solid system for the future. So Tchibo and freiheit.com decided to migrate the whole webshop to the cloud. This makes the system more scalable and flexible as well as preventing potential downtimes in case of unpredicted higher traffic. Ensuring a stable and future-proof system to Tchibo and a reliable shopping experience to their customers.

In 2020 freiheit.com moved Tchibo's webshop to Google Cloud Platform and replaced its monolithic architecture step by step with microservices.

Setting up this new infrastructure in the cloud for a webshop that was built for running on on-premise servers is not easy. But in freiheit.com Tchibo already had a trustworthy partner for over a decade.



*When we looked at what we could achieve with cloud computing, we saw that an on-premises infrastructure would leave us lagging behind in performance, availability and scalability. It was clear that we had to move to the cloud.*

– Frerk Lätari, former Head of Online Technology and Integration at Tchibo.



This is where it all started.

# Meet the key players

## Tchibo GmbH

Tchibo was founded by Max Herz, a tradesman, and his business partner Carl Tchiling in 1949 in Hamburg, Germany.

Their idea? Sending roasted coffee beans via mail to their customers.

This revolutionized the coffee market back then.

Nowadays Tchibo is not only known for their coffee but also for their wide range of weekly changing non-food retail goods. With 900 stores, 24,200 retail depots in ten European countries and more than 10,000 employees.

In the beginning Tchibo's business model was limited to their original idea – they sent coffee beans from their own roastery via mail to their customers. This made the luxury good coffee accessible for everyone. To further improve their customer service, Tchibo opened their first store in Hamburg in 1955, where customers could taste a cup of coffee before buying a package of roasted coffee beans. They expanded fast – just three years later they had 77 branches in Germany. In 1973 they also started selling non-food goods in their stores.

### Tchibo at a glance

**Mission** Selling coffee and a weekly changing range of retail goods

**1949** Founded in Hamburg

**900+** Stores

**24,200** Retail depots

**3,2** Billion euros yearly turnover

**9 Million** of custom large-scale software platforms



In 1994 Tchibo introduced “Themenwelten” to their customers and surprised them with a new wide range of retail goods every week. Driven by the internet boom Tchibo launched their first webshop in 1997. Since then every offered product could be ordered online.

Tchibo is now not only the leading retailer for coffee but also one of the biggest retailers for customer goods in Germany and Europe.

What started in 2009 as a cooperation between Tchibo and freiheit.com to build a new webshop from scratch is now a partnership that has lasted for over 16 years. freiheit.com built a reliable and scalable system that can remain for decades to come.

### **freiheit.com technologies**

freiheit.com was founded in 1999 in Hamburg by Claudia Dietze and Stefan Richter. They have delivered hundreds of large-scale software projects that help their clients create future business models based on software.

They are working for the Who’s Who of European businesses and industries.

From e-commerce to connected cars over retail, healthcare, mobility to logistics and many more industries.

They take on mission impossible projects and build custom large-scale software that has never been built before.

#### **freiheit.com at a glance**

**Mission** Ship great software

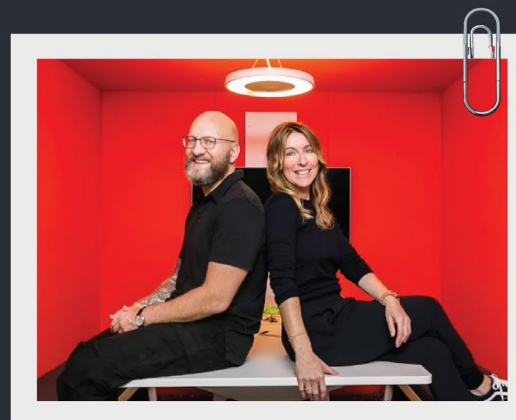
**Always** on time, on budget, on target and on quality

**1999** Founded in Hamburg

**85%** Software Engineers

**2018** Second Engineering Hub in Lisbon

**Hundreds** of custom large-scale software platforms





In 2009 freiheit.com built a new webshop for Tchibo from scratch because their old standardized solution was not scalable anymore. One of the core features was a new Content Management System that enabled Tchibo to reliably launch their weekly changing non-food assortment – their “Themenwelten” – to ensure a smooth customer experience. This new webshop was built as a modular monolith that ran on on-premise servers. Because of its modular monolithic architecture it was already more scalable and flexible than traditional monoliths.

Read more about how freiheit.com built Tchibo’s new webshop completely from scratch, [here](#).





# Defining the problem

When freiheit.com built Tchibo's new webshop in 2009 they ensured that it was a scalable and maintainable system that would be in operation for decades. Building and maintaining such a system also means being able to modernize its technology when needed.

As the webshop still ran on on-premise servers, Tchibo needed to predict increasing webshop traffic months in advance so they could book additional servers manually. If they had higher unpredicted traffic, the demands couldn't be tackled. Tchibo couldn't spontaneously book and set up new servers.

The webshop was built as a monolith. This means when there was higher traffic on e.g. the "cart and check out" part of the webshop they couldn't just scale that specific part. Instead they could only scale up the whole system at once. This was expensive and didn't allow Tchibo to tackle higher demands when needed.

One thing was for sure, Tchibo's webshop needed to remain a solid and stable system they can rely on in the future.

This is why freiheit.com needed to set up a new infrastructure for the webshop.

For this move, Tchibo and freiheit.com partnered with Google to use the advantages of Google Cloud Platform (GCP):

- It allows scaling the application resources up and down based on the demands.
- It offers a variety of integrated databases to use.
- It has built-in security services.
- You only pay for the resources you actually use.
- It has built-in support for continuous integration and delivery.
- It has built-in monitoring and testing features.

To continuously ensure high quality and a future-proof maintainable system [freiheit.com](https://freiheit.com) and Tchibo also decided to split the monolith into a microservice architecture. This allows flexible scaling of the individual microservices.

### **Deep Dive: Monolith vs Microservices**

Tchibo's webshop was built on a modular monolith. Software that is built as a monolith is designed as a single service – built and deployed as a single, unified, self-contained unit. It consists of a database, a user interface and a server-side application. Everything is stored in the same code base – frontend logic as well as background processes and server logic.

Traditional monoliths might become challenging to maintain, because a single change in one component often requires redeploying the entire application.

This is why [freiheit.com](https://freiheit.com) built Tchibo's webshop as a modular monolith which is the evolution of the traditional monolithic architecture. This architecture is structured into independent modules and components, each of them with well-defined boundaries. Every module is independent and isolated.

Microservices are even easier maintainable and extensible than modular monoliths because they break the whole system in small, independent services. Every microservice can be developed, deployed and scaled individually.

# Milestones

## Point of departure & unexpected paths

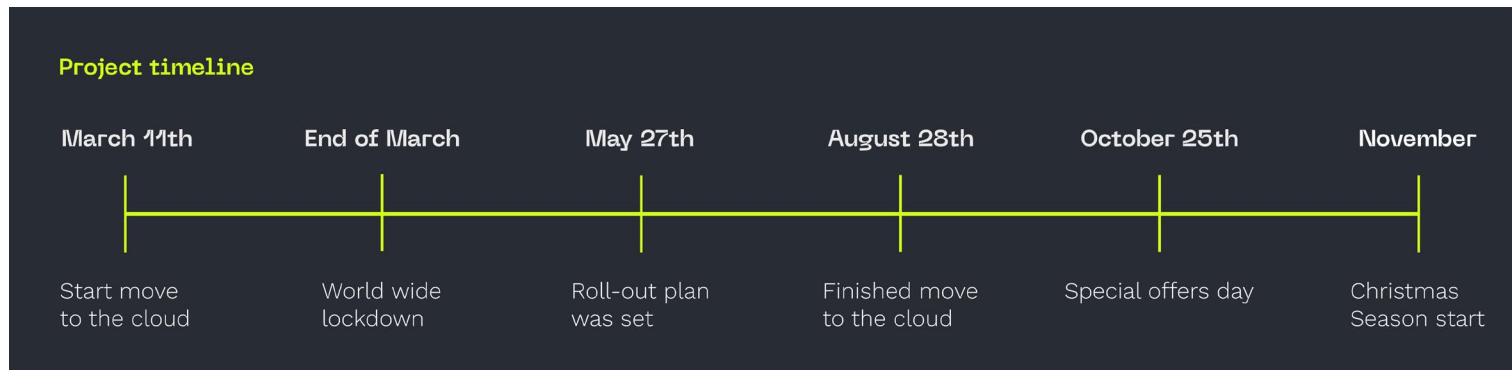
On March 11th, 2020 the decision was made – Tchibo's webshop will be moved to the cloud.

On May 27th, 2020 the roll-out plan was set. Within that timeline, two specific roll-out dates needed to be considered because of higher traffic on those days:

1. Special offer day on October 25th
2. Start of the Christmas season

To guarantee to have everything up and running, the Go-Live date was set for August 28th, 2020. So there was enough time to make sure that everything works in operation.

Shortly after the teams started working everything changed. The Corona Virus spread across the whole world. By the end of March, about half of the world's population was under a lockdown.



The teams had to quickly adapt to working remotely. This included setting up a new digital infrastructure and learning a new way of communication.

freiheit.com and Tchibo knew the project cannot stop – but it was not easy.

Everyone wanted to stick to the plan, no matter what.

Now, with everyone working remotely, they had to figure out how to organize efficient remote meetings and how to continue working together smoothly.

After a short time they worked as efficiently as before and the project kept on going as planned.





## The approach – Moving the webshop to the cloud

As a very first step the engineering teams looked into the code and analyzed it in order to plan how to do the move to the cloud. They quickly realized that they could do a “lift and shift” for the majority of the code because the webshop was a solidly built system. “Lift and shift” means that an existing structure – in this case the modular monolithic architecture of the webshop – is migrated into a new environment. With minimal to no changes.

This is the fastest way to move an existing application to the cloud.

But some parts of the code couldn't just be “lifted and shifted”. These parts needed to be adapted in order to be able to run in the cloud.



**Deep Dive: Technologies that were used for moving to the cloud**

Docker images are the blueprint for running containers, including all the necessary code, libraries, dependencies, and system tools. They are lightweight and portable, ensuring that applications run the same way across different environments. Docker images are built using a Dockerfile – which defines how the application will run inside a container, including instructions for setting up the base image. The Docker image is built from the Dockerfile. Once an image is created, it can be shared, versioned, and stored in a Docker registry.

Kubernetes is an open-source platform for automating the deployment, scaling, and management of containerized applications. While Docker handles the creation and running of individual containers, Kubernetes provides a robust system for orchestrating and managing large numbers of containers across a cluster of machines. It ensures containers are running as expected. It scales applications up or down, and handles failures by automatically restarting or rescheduling containers. Kubernetes organizes containers into pods, which can host one or more containers, providing advanced networking, load balancing, and storage management features.

Moving software to the cloud with Docker and Kubernetes enables a consistent, scalable, and automated deployment of applications across environments, ensuring high availability and efficient resource management.

The “lift and shift” was not everything that needed to be done in order to build a solid and future-proof system in the cloud.

In parallel, the engineering teams also migrated the webshop’s database from Oracle to PostgreSQL.



This switch was mainly motivated by three factors:

**Saving licence costs:** PostgreSQL is open-source and free to use, unlike Oracle which requires expensive licensing fees.

**Compatibility with Google Cloud:** As freiheit.com and Tchibo partnered with Google Cloud for the move to the cloud it was mandatory that the database was as a managed service available in Google Cloud. Oracle was only available as a managed service in Oracle Cloud at that time.

**Effortless development and testing:** It is also easier to set up a PostgreSQL database in a containerized environment than an Oracle one. This makes development and testing simpler.

Moving the webshop to the cloud was done country by country. Due to that, the countries that were already transferred to the cloud were working in the PostgreSQL database, while others were still working on the Oracle database.

Using two different databases at the same time could result in conflicts in the written data. It was really important to ensure that the system would run accordingly all the time to guarantee smooth webshop sales for the customers.

Therefore freiheit.com built a synchronization between both systems. Both databases were used in parallel and needed to contain the same correct data at any time. To reduce the risk of conflicts, freiheit.com restricted write operations on the data to one database. That for example meant that content was still maintained in Oracle but new orders from e.g. Hungary – which webshop was already in the cloud – were written in PostgreSQL. If conflicts still occurred in the data, freiheit.com got alerted immediately to solve the problem.

In that case, action had to be taken quickly to ensure that the webshop continued to run properly. During the whole lift and shift process only a few conflicts occurred. All of them could be tackled immediately – they had no further impact on Tchibo's operations.

### Deep Dive: Migration from Oracle to PostgreSQL

SymmetricDS was used to migrate the data from Oracle to PostgreSQL and continuously sync between the two databases while they were in production use.

The software collects all database operations performed in both instances and executes them on the other side.

The biggest risk was that the same row of a table was changed at the same time on both sides. In that case SymmetricDS would stop the synchronization until the conflict has been resolved manually. To minimize that risk the whole traffic of a side was moved to the cloud environment and the sequences in the cloud got a huge offset so that the primary keys of new data won't conflict.

Nevertheless conflicts could still happen (e.g. if a backend system changes the data of a customer while they are using the eCommerce system) and they did. To detect these cases the logfiles of SymmetricDS were streamed to Datadog (monitoring solution). In Datadog an alert was triggered as soon as an error was detected and a phone call via OpsGenie (on-call-duty solution) was initiated.

Two engineers were always available to resolve those cases.

On August 28th, 2020 the webshop was completely moved to the cloud without any downtime.

**Excatly as planned – on time, on budget, on target and on quality.**

## ***Achievements and results that came with moving Tchibo's webshop to the cloud***

Moving to Google Cloud made the webshop scalable. Additional servers could be booked on short notice. Presumed demands were calculated through expected webshop turnover and traffic. Webshop loading times were faster. That resulted in an improved webshop performance and reliability that improved the shopping experience for each customer. Also deployments could be done with little to no downtime.

**72,000,000** requests per day

**74,000,000** Euros webshop turnover for September 2020

**30%** more turnover than in 2019

**400 ms** “Time to first Byte” (decreased from 800 ms to 400 ms)

**80 ms** Average Product Search Time (decreased from 300 ms to 80 ms)

On Sunday the 25th of October 2020, Tchibo's special offer day took place as planned. With discounts on almost all of their products. Even on this day – with higher demands than on average days – the webshop's performance was stable the whole day.

To quote Tchibo, it even broke the **“world record”** with the **“new record number of orders in one day”**.

From that moment on it was set in stone: that the webshop was ready for the upcoming Christmas Season.

A really great achievement.

## Replacing the monolith with microservices

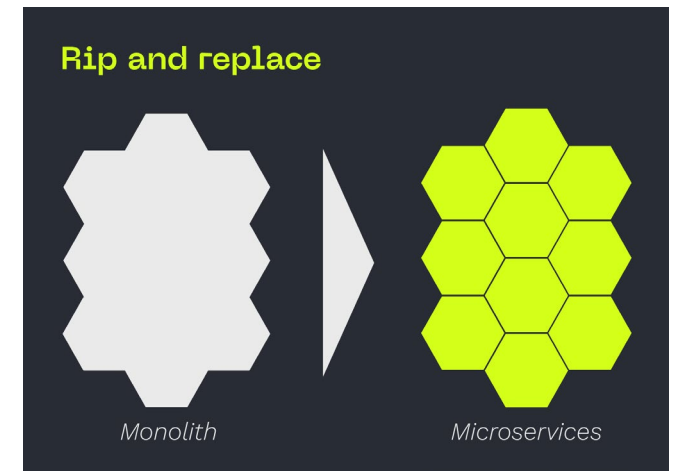
Some modules of the webshop like cart and check out had higher traffic than other modules. With a monolith the entire system needs to scale up when the demands increase for one module. Building the webshop with microservices as the advantage that only the specific module has to be scaled.

So to replace the monolith with microservices the engineering teams broke down the monolithic architecture into small chunks that ran on their own and were independently deployable. The advantage was that every microservice could be adapted, tested, deployed and maintained separately.

The replacement was done with a rip and replace, meaning that the old system was ripped into pieces which then got replaced by microservice.

Tchibo's modular monolith was a complex system with different data sources and webshop pages. And it contained global as well as personalized content. This resulted in multiple connections and dependencies within the webshops structure. Those linkages made the rip and replace a challenge. Detaching one part and replacing it with a microservice was demanding because of that.

During the rip and replace, freiheit.com also continued maintaining the system and adding new features like a coffee subscription to the webshop. With chopping the monolith into microservices parallel development and deployment were possible. It also allowed extensive technical expansions of the individual services during the operation of the webshop.



# How did freiheit.com successfully move Tchibo's webshop to the cloud in three months?

freiheit.com committed to do the move to the cloud in only three months and delivered as promised on time, on budget, on target and on quality.

How did they do it?

## **1. Working in small and efficient teams**

freiheit.com works in small, but efficient engineering teams. One team for each track meaning the different parts of the webshop – like one team for webshop search, cart and check out, front end and back end.

Working in small teams reduces complexity on all ends. Each engineer is responsible for their work, from start to finish. Every member is deeply involved in the creation of the product from the first idea to the final rollout.

## **2. Using backward planning**

All of freiheit.com's teams use a project management tool built by the engineering company for the backward planning of each project. They always start with an ambitious but feasible deadline – the final delivery date. From there on they find out how to fit the project scope into the timeline and then plan backward chronologically from the end of the project with epics and stories to the starting point. An epic defines a feature or increment that can be broken down into stories. The stories are designed as consecutive, dependent steps to reach the epic's goal. This gives every team a clear understanding of what to do in which time schedule.

### 3. Working data driven

freiheit.com uses data and metrics to systematically lead software projects to success. Through data-driven Software Engineering, they significantly increase the delivery reliability of large-scale software projects.

To track the status of the project and the individual team, freiheit.com uses over 40 different metrics. The focus amongst these is on key metrics like:

**Planning Reliability:** This metric describes the percentage of stories that were planned into an iteration at the moment the iteration was started and that were completely finished before the iteration was closed.

**Estimate Deviation:** This metric describes how close the effort estimates were compared to the actual effort needed.

**Development Ratio:** This metric describes how much of the time invested was spent purely on development versus how much time was spent on non-development work. This rate should be at least 60% at all times.

**Service Level Objectives:** This metric describes how good the built service is performing in production.

The data and metrics do not directly indicate what is wrong and what needs to be fixed within the code whenever there is a problem. But they give an indication and can provide warnings if certain thresholds are reached. It is up to the engineers to interpret them and take the correct action to change the metric. The possibility to monitor the metrics over time shows the teams the effectiveness of their derived actions.

This is what freiheit.com calls “find and remove friction”.

The project’s success is rooted in these principles of working.



# Looking back

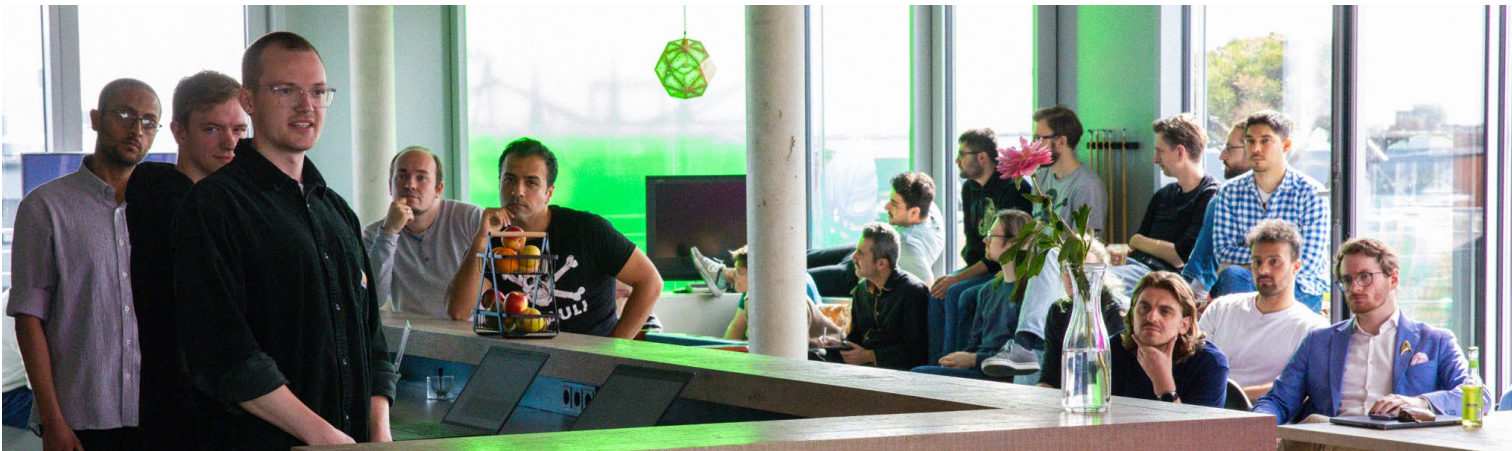
Now Tchibo and freiheit.com can look back on a partnership that has lasted for almost two decades. Together they made Tchibo's webshop successful – the traffic quadrupled over time.

It all started when freiheit.com built Tchibo's webshop from scratch in 2009. This new system enabled Tchibo to efficiently and productively manage and maintain webshop content and all background processes. Every feature runs smoothly all the time, ensuring seamless webshop sales and a great customer experience.

Over the years freiheit.com added new features, rolled out to new countries, moved to the cloud and replaced the monolith with microservices.

freiheit.com built and maintained Tchibo's webshop and all processes in the background for almost two decades.

During all these years the webshop always was a scalable system that can stand the test of time. A system that Tchibo has trusted for years and that they still can trust for decades to come.





# Get in touch with us!

We always love to hear from interesting people. Get in touch with us when you want to join our team or if you need help to make your software project successful.

Send us an email and let the journey begin.

[kontakt@freiheit.com](mailto:kontakt@freiheit.com) →

