



Bringing Reactive Applications to the Java Virtual Machine

Typesafe aids in train safety in the Netherlands

The Typesafe Reactive Platform enabled Xebia to create a solution that allows all parties working on the Dutch rail network to accurately track and account for all cargo moving around the country using mobile devices. In the event of a rail accident, first responders are now armed with accurate information regarding the type of cargo involved and can plan the safest and most expedient way to deal with the situation without putting their lives in unnecessary danger.

About Xebia

Xebia is a highly successful software development and consulting organization. One of Xebia's competence areas is Enterprise Mobile. Enterprise Mobile is about connecting processes across channels and technologies, enabling customers to do business anywhere, anytime. Xebia helps companies succeed in this mobile revolution, making their core systems available on mobile platforms. The tech experts at Xebia deliver cutting edge mobility solutions, which are robust, secure, scalable, & deliver measurable and sustainable improvements in business performance.

The Problem

Every day inside the Netherlands, numerous trains move cargo around the country often carrying dangerous goods.

Shunting - or switching - in railway operations is the process of sorting items of rolling stock into complete train sets or vice-versa. It's a unique job, performed by a person called a shunter. The process of shunting entails moving rail cars from one track to the next within a rail-yard, and at the end of the day, after moving all of the rolling stock around, it's possible that hundreds of rail cars have arrived at and departed from any of the 100+ train yards in The Netherlands.

At any time during the day, the shunter is required to accurately account for all car movements and changes in a paper roster that is handed in at the end of his shift. In this way, there is an accurate picture of what changes have been made on the rail yard. However, a real-time picture of the state of the rolling stock at any point *during* the day or night is *not* available with this legacy system, leaving a substantial amount of time completely unaccounted for. During a calamity within the train yard it takes considerable time to inform the emergency response teams about the exact location of all rail cars and their cargo. This

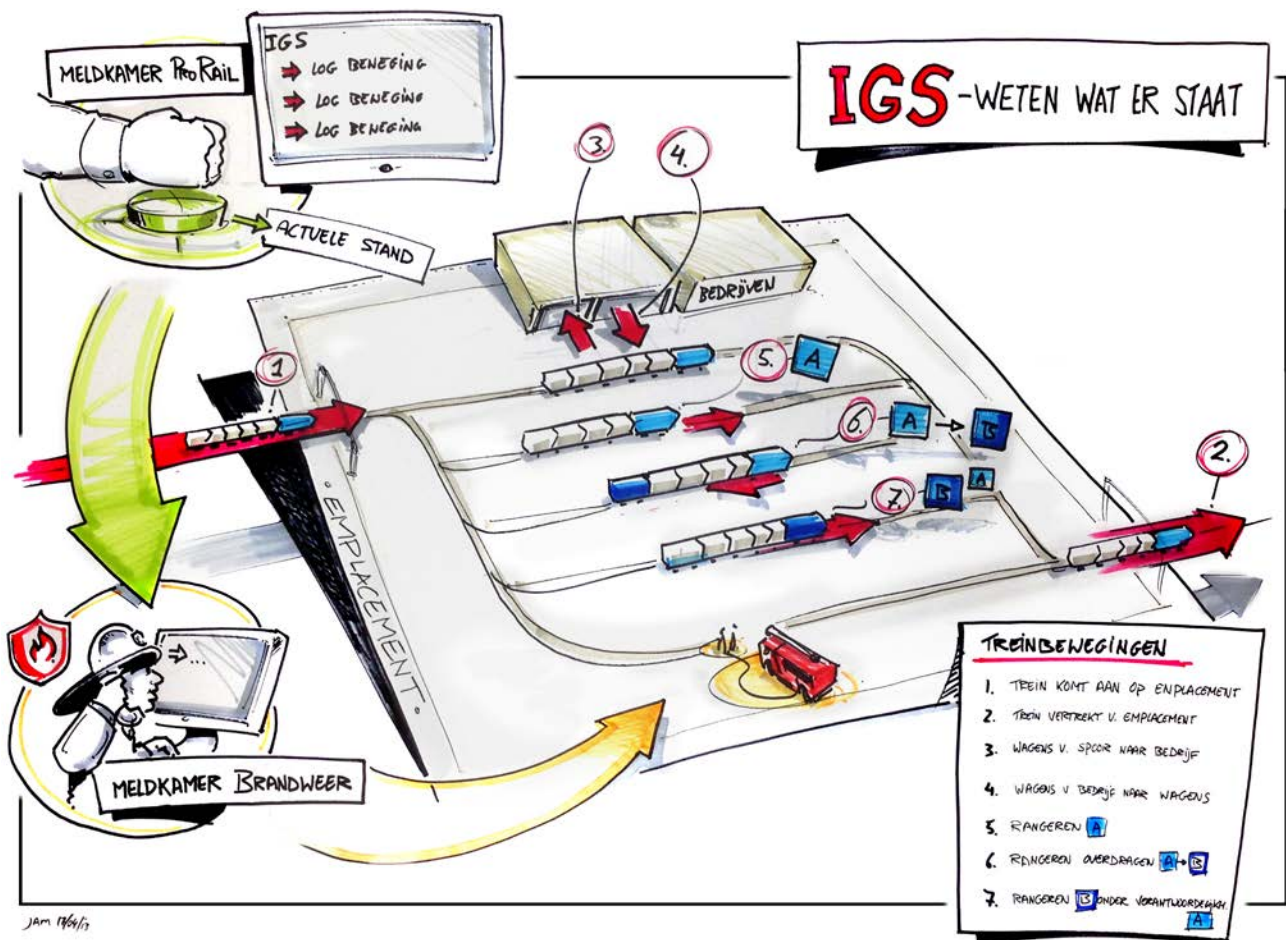
introduces unwanted risks for the first responders, who need to act as soon as possible, depending on the type of goods involved.

During the incident, we didn't know what was burning or where the car stood. We had to climb over several cars, only seeing the faint shimmering of the fire. Only afterwards we learned that all cars contained explosive gasses and liquids.

Fire Brigade Chief

The Solution

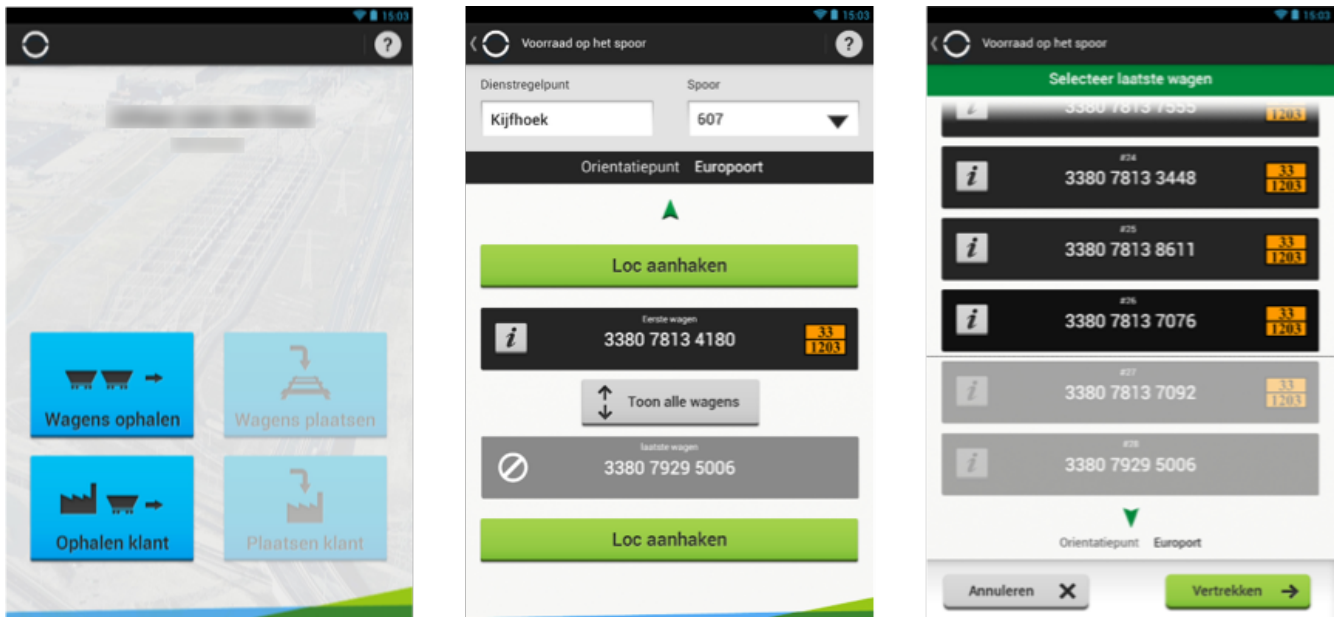
A large-scale program was initiated to address the problem, called IGS (Information Dangerous Substances). Raymond Roestenburg, an Architect with Xebia, lead a small team that was responsible for all mobile development. The team built an entire system, along with fully-fledged iOS and Android applications running on smartphones and tablets that enable shunters to keep a real-time registration that is synchronized to a centralized, coordinating application. The shunter takes his mobile device with him and simply states the track he is going to, the cars he will pick up, and that everything is now "in transit". When he arrives at his destination track, he will enter that track in the device and mark it as "arrived", which subsequently informs the central system of the changes. Since this information is recorded in near real-time, in the case of an emergency, the fire brigade can request a printout of any of the Dutch rail yards, so they can ascertain where the rail cars are, if any carries dangerous cargo and consequently save lives.



During development of this solution, a number of very high profile train accidents occurred in Belgium and Canada, highlighting the need for such a system worldwide.

The Details

Before Xebia joined, the IGS program was already underway and the company Portbase was building a centralized backend system. This main system delivers some of the required functionality via a web-based interface. Raymond proposed that Xebia could very rapidly build a mobile application with an edge-server between the clients and the Portbase backend system. The edge-server's purpose would be to cache reference information for cars and cargo types and furthermore, it would also act as a front-end to the mobile application for registering all of the movements of the rail cars. The Android application was built in 9 weeks and was followed quickly by a subsequent request for an iOS based version.



The edge-server is built with Akka and Spray from Typesafe. Akka is a runtime for building highly concurrent, distributed, and fault tolerant event-driven applications, whereas Spray is a toolkit for building REST/HTTP-based integration layers on top of Scala and Akka. Being asynchronous, Actor-based, fast, lightweight and modular it's a great way to connect Scala applications to the outside world.

Using the powerful combination of Akka and Spray gave the development team tremendous power and flexibility to build the edge-server quickly. Consequently, it was very easy to deploy the application on Heroku for testing and the ultimate production deployment.

Since the backend server was built with legacy technologies, it took longer to create new services for it, however it made sense to keep it in place since it was well tested and stable. The edge-server was effectively the real-time picture of the entire train system. Should Heroku crash or restart for some reason, the entire state of the edge-server is quickly re-read from Amazon S3 and it simply picks up where it left off. Raymond mentions that the system isn't under any huge load, so the scaling aspects of Akka are not strictly required, but what is critical in this system is that every change to the train system is recorded correctly, and in the correct sequence - and only with Akka can that be achieved easily with any level of guarantee.

Conclusion

It took nine weeks to build the first mobile application and the edge-server. The entire project was complete, with fully functioning iOS and Android applications – in production - in eighteen weeks.

Raymond was the sole server-side engineer on this project.

The turnaround time on this project was really tremendous. Also because of the simple model with Akka Actors, it was very easy for me to build in intelligence such as “what has changed on a track?” or “which cars are on the track now, but weren't there before?”

*Raymond Roestenburg,
Architect, Xebia*

*The **Typesafe Reactive Platform** is a modern software platform that makes it easy for developers to build scalable software applications. It combines **Play Framework**, **Akka** runtime, the **Scala** programming language, and robust developer tools in a simple package that integrates seamlessly with existing Java infrastructure. Commercial support and maintenance is available for the Typesafe Reactive Platform through the **Typesafe Subscription**.*