Matthew Gamboa

Module 2.2

11/3/24

The LinkedIn "Operation InVersion" case study is about what happens when tech debt starts to pile up and causes major problems. In 2011, right when Linkedln went public, their main system, called Leo, kept crashing and causing issues that made engineers constantly work on trying to fix the issue. Leo was a big, all-in-one app that handled almost everything on the site, but as LinkedIn grew, Leo couldn’t keep up with all the new traffic. By 2010, they’d already moved some parts out of Leo, but it wasn’t enough as Leo was still hard to work with and a work in progress.

So, LinkedIn’s VP of Engineering, Kevin Scott, decided to try something big. Instead of adding any new features, he stopped all feature work for two months to focus only on fixing their core system. They called this "Operation InVersion." The goal was to break Leo into smaller, separate parts that would be easier to manage and fix. It was a risky choice, especially right after their IPO, because LinkedIn wasn’t delivering anything new to users for two months. But they knew they needed a better foundation if they wanted to keep growing. Most companies don't understand this move but can be very beneficial rather than creating more of a mess and bigger problems teams can't catch up to.

By the end of Operation InVersion, LinkedIn had made some big improvements. Now, engineers could push updates way faster, sometimes a few times a day, instead of waiting weeks. They set up tools to check for bugs early, which cut down on emergency fixes. Josh Clemm, one of the engineering managers, said it made the team feel more flexible and ready for the future. Over time, LinkedIn went from a couple hundred services to more than 750, with a setup that could handle way more users.

Once again, stopping to handle tech debt might feel like a setback, but it’s worth it. By fixing the main problems, LinkedIn set things up for better growth and way fewer issues down the road.