Jordany Gonzalez

January 19, 2025

CSD 380

Module 2.2 Assignment Operation InVersion LinkedIn

The Operation InVersion case at LinkedIn started back in 2003, when the company used a system called Leo to manage its website. At first, Leo seemed like it would work well enough to support the platform, but things changed as LinkedIn started growing fast. The problem was that Leo used a monolithic design, which couldn’t handle the massive amount of data and traffic that LinkedIn was getting. This led to a lot of crashes and made it really hard to release new updates or fix problems. The case study even said, “Leo was often going down in production, it was difficult to troubleshoot and recover, and difficult to release new code.” It became clear that the system just wasn’t cutting it anymore.

The situation got worse because LinkedIn had already announced new features and updates that were coming soon, but with all the issues Leo was having, there was no way they could deliver on those promises. It wasn’t just a matter of fixing small problems—Leo’s design made it super complicated and slow to add anything new. At this point, LinkedIn had no choice but to pause all new development and focus on redoing their entire system from the ground up.

So, LinkedIn decided to stop everything and work on building a better system that could handle the growing demand. They created a whole new set of tools and software that allowed them to update the website much faster, even while it was live. Before, it would take weeks to make changes, but now they could do it multiple times a day if needed. This made a huge difference and helped LinkedIn keep up with its users and stay competitive.

In the end, Operation InVersion wasn’t just about fixing the technology—it was also about rethinking how LinkedIn worked as a company. They focused on making their systems more reliable and scalable so they wouldn’t run into the same problems again. This case shows how important it is to recognize when something isn’t working and have the guts to make big changes, even if it means putting other plans on hold. LinkedIn’s decision to rebuild its system paid off and helped the company stay successful in the long run.

References:

Kim, G., Debois, P., Willis, J., & Humble, J. (2016b). *The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations*. https://dl.acm.org/citation.cfm?id=3044729