Jaci Brown

CSD 380 Mod 6.2

In 2010, David Ashman was the chief software architect of Blackboard Learn, a multi-million-dollar company aimed at online learning for higher education and beyond. Their current system had roots dating back to 1997, including fragments of Perl. Ashman was alerted to declining cycle times and plummeting productivity by internal system data in graphical form.

His attitude toward the problem was to simplify it. The lead times wouldn’t be so long if testing didn’t take so long. Additionally, if testing and reintegration didn’t complicate many layers, it would be easier to make corrections.

David started a re-architecture plan based on the strangler fig pattern. His hopes were that by the end, they could put the original program to rest and rely on the more autonomous and safe to develop in building blocks. The building blocks were favored by the developers, as the more isolated units meant less overlap, coordination, and conflict between code sections. In addition, the errors were more isolated as well, increasing confidence that one wrong move would *not* crash the entire system.

Overall, the modularity of the building blocks that the Blackboard team converted to helped the company improve developer productivity, speed of corrections and feedback, and stability of the entire application.

Reference:

The DevOps Handbook, SE, Ch 13 Case Study: Strangler Fig Pattern at Blackboard Learn (2011)

Referencing Source: “DOES14—David Ashman—Blackboard Learn—Keep Your Head in the Clouds,” YouTube video, 30:43, posted by DevOps Enterprise Summit 2014, October 28, 2014