

Tazmin Somerville

CSD 380 - DevOps

Assignment: Case Study: Strangler Pattern at Blackboard Learn (2011)

Apr 14-20

In Summary

Legacy System Challenges:

- Blackboard Learn's monolithic J2EE codebase (dating back to 1997) led to escalating complexity, error-prone processes, and slow feedback cycles (24–36 hours for integration testing).
- Metrics from 2005–2011 revealed a critical trend: code commits decreased as the monolithic codebase grew, signaling declining developer productivity.

Implementation of the Strangler Fig Pattern:

- In 2012, Chief Architect David Ashman introduced “Building Blocks”, modular components decoupled from the monolith via fixed APIs.
- Developers prioritized working in these modules due to autonomy, faster feedback, and reduced coordination overhead.

Outcomes:

- The monolithic codebase shrank as functionality migrated to Building Blocks.
- The new modules saw exponential growth in code commits and lines of code, reflecting increased productivity and innovation.
- Failures became localized, minimizing system wide disruptions, and faster feedback improved code quality.

Lessons Learned

1. **Modularity Enhances Autonomy:** Decoupling components allows teams to work independently, accelerating development cycles.
2. **Incremental Modernization:** The Strangler Pattern enables gradual legacy system replacement without high risk “big bang” rewrites.
3. **Improved Feedback Loops:** Faster testing and deployment processes boost developer confidence and code quality.

4. **Metrics Drive Decisions:** Tracking commits and codebase size highlighted systemic issues, justifying architectural changes.

Sources:

- Kim, Gene, et al. The DevOps Handbook, Second Edition. IT Revolution Press, 2021.
- Ashman, David. "DOES14—Blackboard Learn—Keep Your Head in the Clouds." *YouTube*, uploaded by DevOps Enterprise Summit, 28 Oct. 2014, <https://www.youtube.com/watch?v=SSmixnMpsl4>.