**Mikaila Steinbrugge**

**Assignment 6.2**  
**February 4, 2025**

In *The DevOps Handbook* (2nd Edition), Chapter 13 explores the **Strangler Fig Pattern** through the case study of Blackboard Learn’s 2011 system overhaul. This pattern is a strategy for gradually modernizing legacy systems by replacing old components with new ones over time, reducing risk and minimizing disruption.

**Key Takeaways**

* **Challenges with Legacy Systems:** Blackboard Learn struggled with a **monolithic architecture** that made it difficult to scale, maintain, and introduce new features efficiently.
* **Using the Strangler Fig Pattern:** To tackle these issues, Blackboard applied the Strangler Fig Pattern, building **new functionalities as independent services** that eventually replaced parts of the legacy system.
* **Incremental Replacement Approach:** Instead of a complete system rewrite, Blackboard developed new services alongside the existing system. Over time, these modernized services **“strangled”** the outdated components, resulting in a fully updated architecture.

**Lessons Learned**

1. **Reducing Risk:** A gradual transition lowers the risks typically associated with large-scale system changes, allowing for testing and validation at each stage.
2. **Supporting Continuous Delivery:** Making small, incremental updates enables **frequent improvements** without disrupting the system.
3. **Enhancing Maintainability:** Shifting from a monolithic system to smaller, modular services makes ongoing maintenance and future upgrades **easier and more scalable**.
4. **Ensuring Business Continuity:** The system remains operational throughout the transition, ensuring **uninterrupted service** for users.

By adopting the Strangler Fig Pattern, Blackboard Learn successfully modernized its platform while avoiding the pitfalls of a **full system overhaul**, demonstrating how this strategy can be an effective approach for managing large-scale refactoring.

Reference:

The DevOps Handbook 2nd. Ed - Chapter 13

<https://www.geeksforgeeks.org/strangler-pattern-in-micro-services-system-design/>