**Module 6.2 Assignment:**

**Strangler Pattern at Blackboard Learn (2011)**

Blackboard Inc. experienced significant challenges with its **Learn product** due to a legacy J2EE codebase dating back to 1997, which hindered development efficiency and productivity. By 2011, the company reported annual revenues of approximately **$650 million**. The chief architect, **David Ashman**, noted complications arising from an increasingly complex system, where build, integration, and testing processes became error-prone, leading to extended lead times for feedback—ranging from **24 to 36 hours**.

Graphs from their source code repository revealed a troubling trend. While the **lines of code** in the monolithic repository increased, the **number of code commits** decreased, indicating difficulties in implementing changes. This situation prompted Ashman to initiate a **code re-architecting project** using the **strangler fig pattern**.

In 2012, the team introduced **Building Blocks** and modular components, allowing developers to work independently from the monolithic codebase through fixed APIs. This shift enabled greater autonomy and reduced the need for constant communication among teams. As developers transitioned their code into the Building Block modules, the size of the monolith decreased.

The implementation resulted in an **exponential growth** in code commits within the Building Blocks codebase, enhancing developer productivity and safety by localizing mistakes to smaller modules rather than affecting the entire system. Ashman highlighted that this modular architecture **improved code quality** and faster feedback loops, fostering a more efficient development environment.