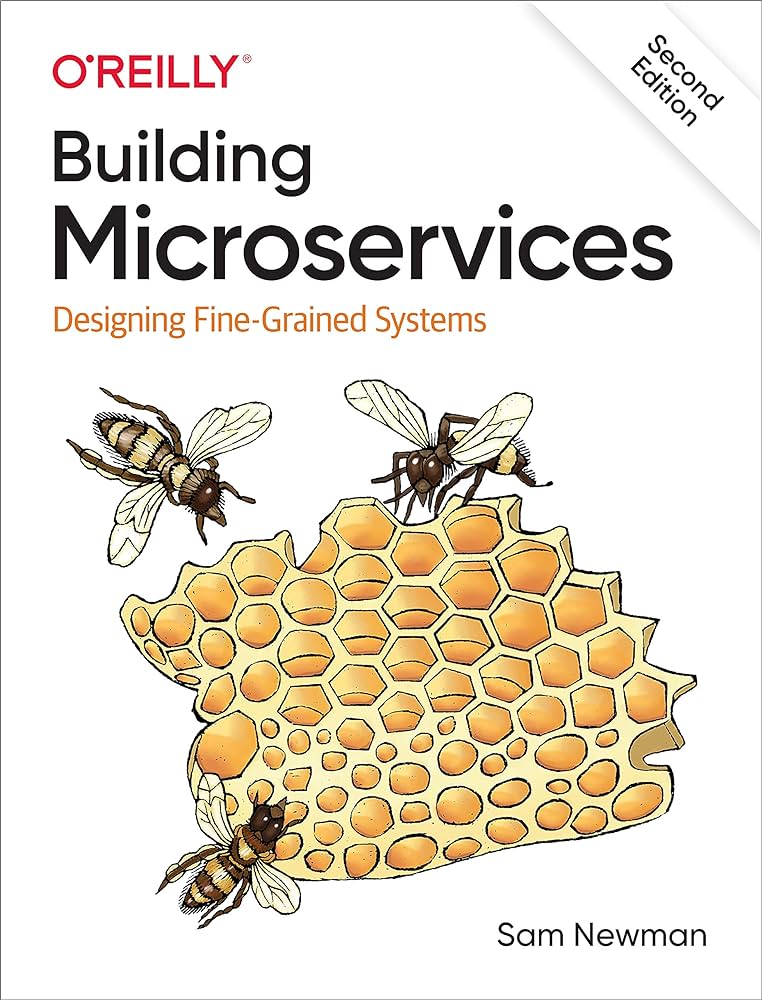
At work we’ve started to use microservices for some of our projects, moving away from the monolith architecture. To learn more about this, I decided to read “Building Microservices by Sam Newman”.



With monolithic architectures, all processes are coupled and run as a single service. This means if one process of the application is to experience a sudden spike of demand, the entire architecture must be scaled. Adding or improving a monolithic application’s features becomes more complex as the codebase grows. The complexity limits experimentation and makes it difficult to implement new ideas. Tightly coupled processes increase the impact of a single process failure.

With a microservice architecture, an application is built as independent components that run each application process as a service. These services communicate via a well-defined interface using lightweight APIs. Services are built for business capabilities and each service performs a single function. Because they run independently, each service can be updated, deployed, and scaled to meet demand for specific functions of an application.

Benefits of microservices include:

* Flexible scaling: each microservice can independently be scaled to meet demand.
* Easy Deployment: Microservices enable continuous integration and continuous delivery, making it easy to try out new ideas and to roll back if something doesn’t work.
* Technological Freedom: Microservices architectures don’t follow a “one size fits all” approach. Teams have the freedom to choose the best tool to solve their specific problems. Therefore, teams building microservices can choose the best tool for each job.