Hi everyone,

For my topic I chose, Conway’s Law. The reason I got drawn to it might be the silliest, but the name triggers my memory of Murphy’s Law (Anything that can go wrong, will go wrong), hence I chose this.

Conway’s Law highlights an interesting link between how teams communicate and how systems are designed, suggesting that an organization’s structure influences the structure of the software it builds. Simply put, if a company has separate teams for frontend and backend, for instance, the product may reflect that divide with distinct frontend and backend components.

There are some clear advantages to Conway’s Law. For example, cross-functional teams in an agile organization often lead to more modular and flexible systems. When teams are aligned, collaboration improves, allowing faster iteration cycles and better adaptability to changes. This approach is especially helpful for complex projects where quick adjustments and close collaboration are key.

However, Conway’s Law also comes with some downsides. As companies grow, their communication structures can become fragmented, leading to siloed designs that make it harder to integrate or adapt. For example, if departments are isolated, the software may become equally disconnected, making it challenging to implement cohesive changes across the system. Plus, when organizations undergo restructuring, it can disrupt the product architecture, leading to technical debt and misaligned priorities.

In the end, Conway’s Law serves as a reminder to think about how team organization impacts system design. By aligning team structures with product goals, companies can create systems that are easier to develop and maintain rather than being hindered by organizational divides.

<https://www.splunk.com/en_us/blog/learn/conways-law.html>

<https://effectiveagile.com/agile/what-is-conways-law/>