

I've broken the decision down into the different areas that we might consider.

Area	Monolithic	Microservice
Deployment	This architecture allows you to deploy once and then customize your solution based on current changes. But if something goes wrong, the whole project will break down.	Deployment is a complex process in microservices. Each microservice must be implemented separately, which extends the implementation process. If something goes wrong, only one microservice will be damaged, and it will be easier to fix it.
Maintenance	Maintenance in a monolithic architecture requires an IT team that specializes in multiple platforms, like Pascal, .NET, java, or DB2. Finding errors and making changes takes a lot of time in the monolithic. However, testing itself is simple and will be done in one go.	Maintenance in microservices is faster than in monolithic. Smaller services are also easy to test, saving programmers time. Over time, it increases efficiency and saves money.
Reliability	If it comes to reliability, the monolith has no chance against microservices. If something goes wrong in monolithic architecture, it can stop the whole structure. Meanwhile, in microservices architecture, breaking one service will not cause critical problems in the overall application.	Microservices are stable and mostly reliable. Breaking one part only affects that element, while the others remain intact. Such flexibility allows for a fast pace of development and the introduction of changes in one function without interfering with others.
Scalability	Scalability in monolithic architecture is difficult to achieve due to the size and scale of the structure. This option is hard to upgrade.	In the case of microservices, scalability is much easier because we can scale only those parts that require more resources.

Like everything, these two technologies both have their advantages and disadvantages. Monolithic architecture is a solid solution for simple eCommerce applications or blog platforms, especially when ongoing changes and development are not predicted. Microservices have grown to become a better fit for complex applications and are the modern solution in a time when constant improvement and development of sites and service is the norm.