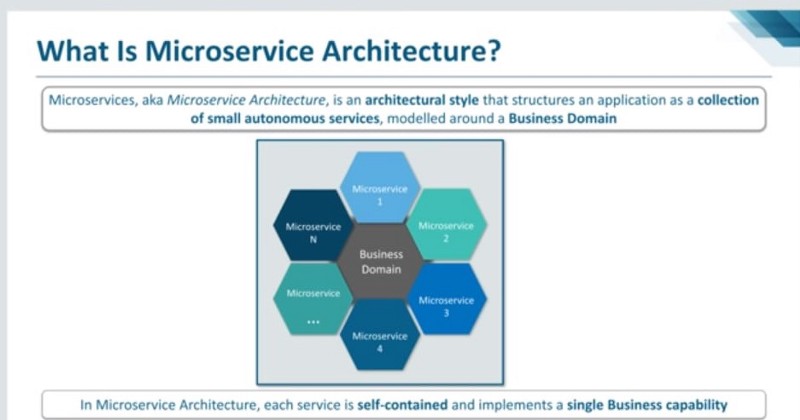
# What is Microservices ?

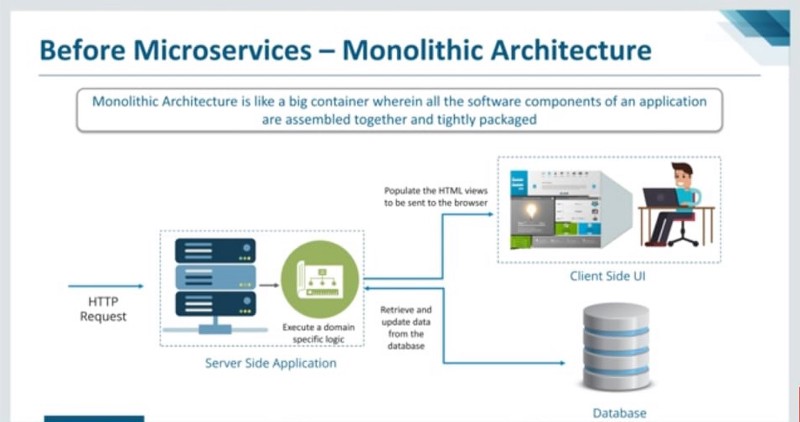


Microservices is a hot topic in software development circles these days. And for some very good reasons.

Put simply, the traditional way of building enterprise applications — using a [monolithic approach](https://en.wikipedia.org/wiki/Monolithic_application) — has become problematic as applications get larger and more complex. So developers are turning to a microservices software development architecture, in which applications are structured as collections of loosely coupled services. This makes them easier to build, and — more importantly — much easier to expand and scale.

Let’s take a closer look at how a microservices approach differs from a monolithic one, and examine their relative strengths and weaknesses. Before moving further first we understand the Monolithic architecture in detail, in order to understand the microservices better and later we will also differentiate among them so that you are about to be a pro in the topic.

# What is Monolithic Architecture??



A monolithic architecture is the traditional unified model for the design of a software program. Monolithic, in this context, means composed all in one piece. Monolithic software is designed to be self-contained; components of the program are interconnected and interdependent rather than loosely coupled as is the case with modular software programs. In a tightly-coupled architecture, each component and its associated components must be present in order for code to be executed or compiled.

