aws

Modern Application Development: Modern compute

Modern compute

When you're building an application, you generally have three options for compute: instances, containers, and Lambda. What you choose and how you choose it depends on how much you care about flexibility versus simplicity, because there are tradeoffs. We define "simplicity" as seamless integration with other services and features and flexibility as the ability to make more decisions about configuration. Increasingly we're seeing our customers choose containers and Lambda for compute. Containers offer excellent portability and flexibility over an application's settings. Lambda offers the most simplicity—the only code you write is business logic.

Build and run apps without thinking about servers

Spend time focusing on innovating your application instead of managing infrastructure. You no longer need to provision or maintain servers, operating systems, or software. This means you can focus on writing business logic that provides differentiated value to your customers and business. Serverless technologies automatically scale by unit of work and have built-in availability and fault tolerance, so you can build customer-ready applications, right from the start. You can build modern applications with a lower cost of ownership because you pay for execution duration, or consistent throughput, not for server unit.

About containers

Containers provide a standard way to package your application's code, configurations, and dependencies into a single object. Containers share an operating system installed on the server and run as resource-isolated processes, ensuring quick, reliable, and consistent deployments, regardless of environment. Containers package your code with the configuration files and dependencies it needs to consistently run in any environment. Containers also provide process isolation that lets you granularly set CPU and memory utilization for better use of compute resources.

Let's focus on compute for now

AWS Lambda
Serverless event-driven code execution

Short-lived
All language runtimes
Data source integrations

AWS Fargate
Serverless compute engine for containers

Long-running
Bring existing code
Fully managed orchestration

aws

Modern Application Development:
Modern compute

The power of "And": Serverless and containers

For building net-new applications, you should consider using serverless technologies such as AWS Lambda and AWS Fargate. With serverless, you no longer need to provision or maintain servers, operating systems, or software. This means you can focus on writing business logic that provides differentiated value to your customers and business.

For modernizing legacy apps and migrating these to the cloud, you should consider using containers. Containers help you provide a consistent packaging and deployment environment, which facilitates on-premises portability and hybrid scenarios as you build your cloud migration strategy. Containers also provide complete control of the compute environment enabling you to have more granular control of your infrastructure to ease the complexity of migrating your apps to the cloud.

What is serverless?

- No infrastructure provisioning, no management
- Automatic scaling
- Pay for value
- Highly available and secure

According to research from IDC, organizations that adopt serverless technologies reduce their five-year operating cost by 60% while increasing compute deployment efficiency by 89%.

Ready to see how it works?

Visit us to learn more about how to leverage modern compute practices.