



---

## Why AWS?

---



# TechData-Infinity-Devops with MultiCloud



## 2. Why AWS?

This course is gonna cover all the essential information that you need to understand, to be comfortable discussing AWS, to know why it's beneficial to your business.

AWS offers a massive range of services for every business, starting with basic elements, like compute, storage, and network security tools, through complex solutions like blockchain, machine learning, or artificial intelligence, and robot development platforms, all the way through very specialized tool sets, like video production management systems, and orbital satellites you can rent by the minute.

All that, however, is way more than we have time to cover in a foundational class like this one. So let's simplify the conversation by starting with the fundamental cloud compute model.

I know it can be complicated, so let's take a look at our coffee shop.

This coffee shop is going to give us some real world examples to help you understand why AWS can change the way your IT operates.

Let's make Morgan the server, the barista. And I am the client, the customer. I make a request. In this case, it is for coffee. Now in the computing world, the request could be anything. It could be rain pattern analysis in South Africa, or the latest x-rays of your knee, or videos of kittens. Whatever is the business, basically a customer makes a request, and with permissions, the server responds to that request. All I want is a caffeinated beverage.

Morgan represents the server part of the client-server model. **In AWS, she would be called an Amazon Elastic Compute Cloud, or EC2, an EC2 instance, a virtual server.** So from an architectural point of view, the transaction we did is really simple to explain. I, the user, made a request to Morgan, the server. Morgan validated that the request was legitimate, in this case, did I give her money? Then she returned a response, which in this case, is a berry blaster with extra caramel shots.

Now in the real world, applications can get more complicated than just a single transaction with a single server. In a business solution that is more mature, it can get beautifully complex.

To avoid this complexity, we're going to start simple. We will build this discussion out so that it is easy for anyone to understand how these concepts build on each other. So, by the end, those complex concepts, they'll be easy to understand. Let's start with a key concept to AWS, and that is, you only pay for what you use.

This principle makes sense when you run a coffee shop. Employees are only paid when they're in the store working. If Rudy and Morgan are off the clock, well then they don't get paid. The store owner simply decides how many baristas are needed and then just pays for the hours they work. For example, the coffee shop is about to release a new drink, the Pumpkin Monster Spice. In anticipation of this launch, you could always staff your shop with a dozen baristas all day

# TechData-Infinity-Devops with MultiCloud



long, just in case you suddenly get an unexpected rush at some point in the day. Only, let's be honest. For most of your day, you don't have near enough customers to justify paying for all those employees.

And yet, this is exactly what happens in an on-premises data center. You can't just snap your fingers and triple your capacity. At AWS, you don't pre-pay for anything. And you don't have to worry about capacity constraints.

When you need instances, or baristas, you just click a button, and you have them. And when you don't need them, another click, and they go away, and you stop paying for them. The same way you don't pay for employees for hours that they're not working.

So, pay for what you need, becomes the first key value of many for running your business on AWS. And that is really why we're here, to help you understand how AWS is built to help you run your business better.

## What is a client-server model?

You just learned more about AWS and how almost all of modern computing uses a basic client-server model. Let's recap what a client-server model is.

### Client



### Server



In computing, a client can be a web browser or desktop application that a person interacts with to make requests to computer servers. A server can be services such as Amazon Elastic Compute Cloud (Amazon EC2), a type of virtual server.

For example, suppose that a client makes a request for a news article, the score in an online game, or a funny video. The server evaluates the details of this request and fulfills it by returning the information to the client.

# TechData-Infinity-Devops with MultiCloud



AWS Market Share–

## IaaS/PaaS market (Source: Jefferies)

Cloud vendor	Annualized revenue	% of market	Year-over-year growth
Amazon Web Services	\$18.34 billion	51%	42%
Microsoft Azure	\$6.17 billion	17%	89%
IBM Cloud	\$4.03 billion	11%	22%
Google Cloud Platform	\$2.05 billion	6%	125%
Alibaba Cloud	\$1.79 billion	5%	92%
Salesforce	\$1.78 billion	5%	31%
Oracle Cloud	\$1.59 billion	4%	20%
<b>Subtotal</b>	\$35.75 billion	86%	54%
<b>Total Gartner estimate</b>	\$41.79 billion	100%	33%