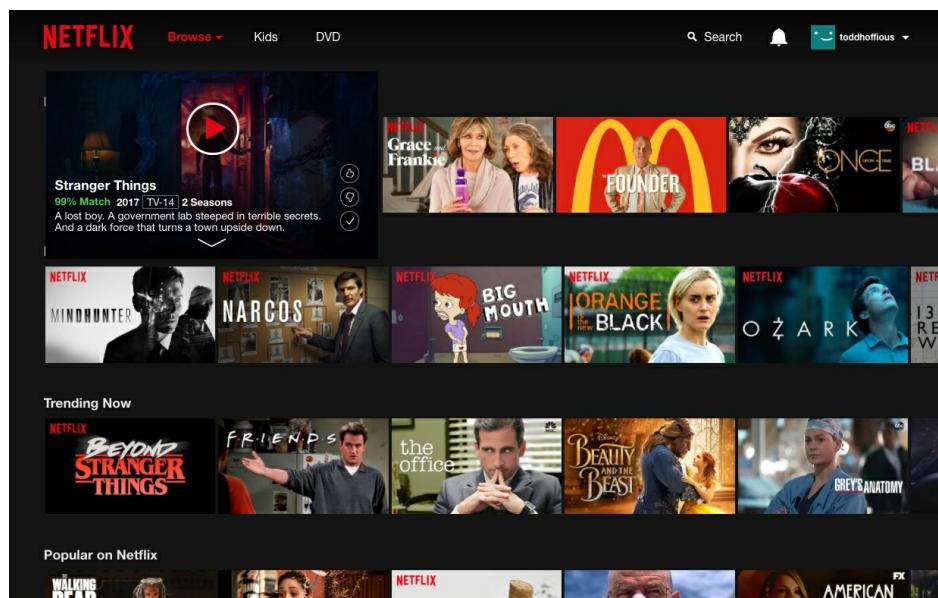


NETFLIX: WHAT HAPPENS WHEN YOU PRESS PLAY?



Netflix seems so simple. Press play and video magically appears. Easy, right? Not so much.



Given our discussion in the *What is Cloud Computing?* chapter, you might expect Netflix to serve video using AWS. Press play in a Netflix application and video stored in S3 would be streamed from S3, over the internet, directly to your device.

A completely sensible approach...for a much smaller service.

But that's not how Netflix works at all. It's far more complicated and interesting than you might imagine.

To see why let's look at some impressive Netflix statistics for 2017.

- Netflix has more than 110 million subscribers.
- Netflix operates in more than 200 countries.
- Netflix has nearly \$3 billion in revenue per quarter.
- Netflix adds more than 5 million new subscribers per quarter.
- Netflix plays more than 1 billion hours of video each week. As a comparison, YouTube streams 1 billion hours of video *every day* while Facebook streams 110 million hours of video every day.
- Netflix played 250 million hours of video on a single day in 2017.
- Netflix accounts for over 37% of peak internet traffic in the United States.
- Netflix plans to spend \$7 billion on new content in 2018.

What have we learned?

Netflix is huge. They're global, they have a lot of members, they play a lot of videos, and they have a lot of money.

Another relevant factoid is Netflix is subscription based. Members pay Netflix monthly and can cancel at any time. When you press play to chill on Netflix, it had better work. Unhappy members unsubscribe.

We're Going Deep

Netflix is a terrific example of all the ideas we've talked about, which is why this chapter goes into a lot more detail than the other cloud services we've covered.

One big reason for diving deeper into Netflix is they make much more information available than other companies.

Netflix holds *communication* as a central cultural value. Netflix more than lives up to its standards.

In fact, I'd like to thank Netflix for being so open about their architecture. Over the years, Netflix has given hundreds of talks and written hundreds of

articles on the inner-workings of how they operate. The whole industry is better for it.

Another reason for going into so much detail on Netflix is Netflix is just plain fascinating. Most of us have used Netflix at one time or another. Who wouldn't love peeking behind the curtain to see what makes Netflix tick?

Netflix operates in two clouds: AWS and Open Connect.

How does Netflix keep their members happy? With the cloud of course. Actually, Netflix uses two different clouds: AWS and Open Connect.

Both clouds must work together seamlessly to deliver endless hours of customer-pleasing video.

The three parts of Netflix: client, backend, content delivery network (CDN).

You can think of Netflix as being divided into three parts: the client, the backend, and the content delivery network (CDN).

The *client* is the user interface on any device used to browse and play Netflix videos. It could be an app on your iPhone, a website on your desktop computer, or even an app on your Smart TV. Netflix controls each and every client for each and every device.

Everything that happens before you hit *play* happens in the *backend*, which runs in AWS. That includes things like preparing all new incoming video and handling requests from all apps, websites, TVs, and other devices.

Everything that happens after you hit *play* is handled by Open Connect. Open Connect is Netflix's custom global content delivery network (CDN). Open Connect stores Netflix video in different locations throughout the world.

When you press play the video streams from Open Connect, into your device, and is displayed by the client. Don't worry; we'll talk more about what a CDN is a little later.

Interestingly, at Netflix they don't actually say *hit play on video*, they say *clicking start on a title*. Every industry has its own lingo.

By controlling all three areas—client, backend, CDN—Netflix has achieved

complete vertical integration.

Netflix controls your video viewing experience from beginning to end. That's why it just works when you click play from anywhere in the world. You reliably get the content you want to watch when you want to watch it.

Let's see how Netflix makes that happen.

In 2008 Netflix Started Moving to AWS

Netflix launched in 1998. At first they rented DVDs through the US Postal Service.



*Here's what Netflix looked like in 1999 when they rented only DVDs.
(Netflix)*

But Netflix saw the future was on-demand streaming video.

In 2007 Netflix introduced their streaming video-on-demand service that allowed subscribers to stream television series and films via the Netflix website on personal computers, or the Netflix software on a variety of supported platforms, including smartphones and tablets, digital media players, video game consoles, and smart TVs.



*Here's what Netflix looked like in 2008 when they introduced streaming.
(Netflix)*

On a personal note, that streaming video-on-demand was the future might seem obvious. And it was. I worked at a couple of startups that tried to make a video-on-demand product. They failed.

Netflix succeeded. Netflix certainly executed well, but they were late to the game, and that helped them. By 2007 the internet was fast enough and cheap enough to support streaming video services. That was never the case before. The addition of fast, low-cost mobile bandwidth and the introduction of powerful mobile devices like smart phones and tablets, has made it easier and cheaper for anyone to stream video at any time from anywhere. Timing is everything.

Netflix Began by Running their Own Datacenters

EC2 was just getting started in 2007, about the same time Netflix's streaming service started. There was no way Netflix could have launched using EC2.

Netflix built two datacenters, located right next to each other. They experienced all the problems we talked about in earlier chapters.

Building out a datacenter is a lot of work. Ordering equipment takes a long time. Installing and getting all the equipment working takes a long time. And as soon they got everything working they would run out of capacity, and the whole process had to start over again.

The long lead times for equipment forced Netflix to adopt what is known as a *vertical scaling* strategy. Netflix made big programs that ran on big computers. This approach is called building a *monolith*. One program did everything.

The problem is when you're growing really fast like Netflix; it's very hard to make a monolith reliable. And it wasn't.

A Service Outage Caused Netflix to Move to AWS

For three days in August 2008, Netflix could not ship DVDs because of corruption in their database. This was unacceptable. Netflix had to do something.

The experience of building datacenters had taught Netflix an important lesson —they weren't good at building datacenters.

What Netflix was good at was delivering video to their members. Netflix would rather concentrate on getting better at delivering video rather than getting better at building datacenters. Building datacenters was not a competitive advantage for Netflix, delivering video is.

At that time, Netflix decided to move to AWS. AWS was just getting established, so selecting AWS was a bold move.

Netflix moved to AWS because it wanted a more reliable infrastructure. Netflix wanted to remove any single point of failure from its system. AWS offered highly reliable databases, storage and redundant datacenters. Netflix wanted cloud computing, so it wouldn't have to build big unreliable monoliths anymore. Netflix wanted to become a global service without building its own datacenters. None of these capabilities were available in its old datacenters and never would be.

A reason Netflix gave for choosing AWS was it didn't want to do any *undifferentiated heavy lifting*. Undifferentiated heavy lifting are those things that have to be done, but don't provide any advantage to the core business of providing a quality video watching experience. AWS does all the undifferentiated heavy lifting for Netflix. This lets Netflixians focus on providing business value.

It took more than eight years for Netflix to complete the process of moving from their own datacenters to AWS. During that period Netflix grew its number of streaming customers eightfold. Netflix now runs on several hundred thousand EC2 instances.

Netflix is More Reliable in AWS

It's not like Netflix has never experienced down time on AWS, but on the whole, its service is much more reliable than it was before.

You don't see complaints like this very often anymore:



or this:



Netflix is so reliable now because they've taken extraordinary steps to make their service reliable.

Netflix operates out of three AWS regions: one in North Virginia, one in Portland Oregon, and one in Dublin Ireland. Within each region, Netflix operates in three different availability zones.

Netflix has said there are no plans to operate out of more regions. It's very expensive and complicated to add new regions. Most companies operate out of just one region, let alone two or three.

The advantage of having three regions is that any one region can fail, and the other regions will step in handle all the members in the failed region. When a region fails, Netflix calls this *evacuating* a region.

Let's use an example. Let's say you're watching a new *House of Cards* episode in London England. Because it's closest to London, chances are your Netflix device is connected to the Dublin region.

What happens if the entire Dublin region fails? Does that mean Netflix should stop working for you? Of course not!

Netflix, after detecting the failure, redirects you to Virginia. Your device would now talk to the Virginia region instead of Dublin. You might not even notice there was a failure.

How often does an AWS region fail? Once a month. Well, a region doesn't actually fail every month. Netflix runs monthly tests. Every month Netflix causes a region to fail on purpose just to make sure its system can handle region level failures. A region can be evacuated in six minutes.

Netflix calls this their *global services model*. Any customer can be served out of any region. This is amazing. And it doesn't happen automatically. AWS has no magic sauce for handling region failures or serving customers out of multiple regions. Netflix has done all this work on its own. Netflix is a pioneer in figuring out how to create reliable systems using multiple regions. I'm not aware of any other company that goes to these lengths to make their service so reliable.

Another advantage of being in these three regions is that it gives Netflix world-wide coverage. Netflix ran some tests and found if you use a Netflix application anywhere in the world, you'll get fast service from one of these three regions.

Netflix Saves Money in AWS

This may surprise a lot of people, but AWS is cheaper for Netflix. The cloud costs per streaming view ended up being a fraction of the cost of its old datacenters.

Why? The elasticity of the cloud.

Netflix could add servers when it needed them and return them when it didn't. Rather than have a lot of extra computers hanging around doing nothing just to handle peak load, Netflix only had to pay for what was needed, when it was needed.

All the stuff we talked about in the *What is Cloud Computing?* chapter.

What Happens in AWS Before you Press Play?

Anything that doesn't involve serving video is handled in AWS.

This includes scalable computing, scalable storage, business logic, scalable distributed databases, big data processing and analytics, recommendations, transcoding, and hundreds of other functions.

Don't worry, you don't need to understand what all those things are, but since you may find it interesting, I'll explain them briefly.

Scalable computing and scalable storage.

Scalable computing is EC2 and *scalable storage* is S3. Nothing new for us here.

Your Netflix device—iPhone, TV, Xbox, Android phone, tablet, etc.—talks to a Netflix service running in EC2.

View a list of potential videos to watch? That's your Netflix device contacting a computer in EC2 to get the list.

Ask for more details about a video? That's your Netflix device contacting a computer in EC2 to get the details.

It's just like all the other cloud services we've talked about in the book.

Scalable distributed database.

Netflix uses both DynamoDB and Cassandra for their distributed databases. Not that these names should mean anything to you, they're just high-quality database products.

Database. A database stores data. Your profile information, your billing information, all the movies you've ever watched, all that kind of information is stored in a database.

Distributed. Distributed means the database doesn't run on one big computer, it runs on many computers. Your data is copied to multiple computers so if one or even two computers holding your data fail, your data will be safe. In fact, your data is copied to all three regions. That way, if a region fails your data will be there when the new region is ready to start using it.

Scalable. Scalable means the database can handle as much data as you ever want to put into it. That's one major advantage of being a distributed database. More computers can be added as necessary to handle more data.

Big data processing and analytics.

Big data simply means there's a lot of data. Netflix collects a lot of information. Netflix knows what everyone has watched when they watched it and where they were when they watched. Netflix knows which videos members have looked at but decided not to watch. Netflix knows how many times each video has been watched...and a lot more.

Putting all the data in a standard format is called *processing*.

Making sense of all that data is called *analytics*. Data is analyzed to answer specific questions.

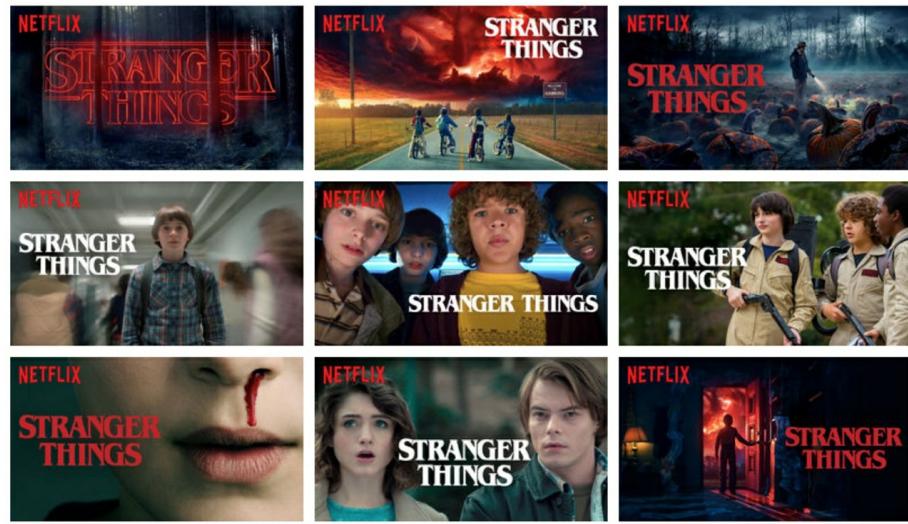
Netflix personalizes artwork just for you.

Here's a great example of how Netflix entices you to watch more videos using its data analytics capabilities.

When browsing around looking for something to watch on Netflix, have you noticed there's always an image displayed for each video? That's called the *header image*.

The header image is meant to intrigue you, to draw you into selecting a video. The idea is the more compelling the header image, the more likely you are to watch a video. And the more videos you watch, the less likely you are to unsubscribe from Netflix.

Here's an example of different header images for *Stranger Things*:



Netflix

You might be surprised to learn the image shown for each video is selected specifically for you. Not everyone sees the same image.

Everyone used to see the same header image. Here's how it worked. Members were shown at a random one picture from a group of options, like the pictures in the above *Stranger Things* collage. Netflix counted every time a video was watched, recording which picture was displayed when the video was selected.

For our *Stranger Things* example, let's say when the group picture in the center was shown, *Stranger Things* was watched 1,000 times. For all the other pictures, it was watched only once each.

Since the group picture was the best at getting members to watch, Netflix would make it the header image for *Stranger Things* forever.

This is called being *data-driven*. Netflix is known for being a data-driven company. Data is gathered—in this case, the number of views associated with each picture—and used to make the best decisions possible—in this case, which header image to select.

Clever, but can you imagine doing better? Yes, by using more data. That's

the theme of the future—solving problems by learning from data.

You and I are likely very different people. Do you think we are motivated by the same kind of header image? Probably not. We have different tastes. We have different preferences.

Netflix knows this too. That's why Netflix personalizes all the images they show you. Netflix tries to select the artwork highlighting the most relevant aspect of a video to you. How do they do that?

Remember, Netflix records and counts everything you do on their site. They know which kind of movies you like best, which actors you like the most, and so on.

Let's say one of your recommendations is the movie *Good Will Hunting*. Netflix must choose a header image to show you. The goal is to show an image that lets you know about a movie you'll probably be interested in. Which image should Netflix show you?

If you like comedies, Netflix will show you an image featuring Robin Williams. If you prefer romantic movies, Netflix will show you an image Matt Damon and Minnie Driver poised for a kiss.



By showing Robin Williams, Netflix is letting you know there's likely to be humor in the movie and because Netflix knows you like comedies, this video is a good match.

The Matt Damon and Minnie Driver image conveys a completely different message. If you're a comedy fan and saw this image, you might skip right on

by.

That's why selecting the right header image is so important. It sends a strong personalized signal indicating what a movie is about.

Here's another example, *Pulp Fiction*.



If you've watched a lot of movies starring Uma Thurman, then you're likely to see the header image featuring Uma. If you've watched a lot of movies starring John Travolta, then you're likely to see the header image featuring John.

Can you see how choosing the best possible personalized artwork might make you more likely to watch a video?

Netflix appeals to your interests when selecting artwork, yet Netflix doesn't want to lie to you either. They don't want to show a clickbait image just to get you to watch a video you may not like. There's no incentive in that. Netflix isn't paid per video watched. Netflix tries to *minimize regret*. Netflix wants you to be happy with the videos you watch, so they pick the best header images they can.

This is just one small example of how data analysis is used by Netflix. Netflix uses these kind of strategies everywhere.

Recommendations.

Usually Netflix will show you only 40 to 50 video options, yet they have many thousands of videos available.

How does Netflix decide? Using machine learning.

That's part of the *big data processing and analytics* we just talked about. Netflix looks at its data and predicts what you'll like. Nobody else's Netflix screen looks like yours. Everything you see on a Netflix screen is chosen specifically for you using machine learning.

Transcoding From Source Media to What You Watch

Here's where we start transitioning into how video is handled by Netflix.

Before you can watch a video on your favorite device of choice, Netflix must convert the video into a format that works best for your device. This process is called *transcoding* or *encoding*.

Transcoding is the process that converts a video file from one format to another, to make videos viewable across different platforms and devices.

Netflix encodes all its video in AWS on as many as 300,000 CPUs at one time. That's larger than most super computers!

The source of source media.

Who sends video to Netflix? Production houses and studios. Netflix calls this video *source media*. The new video is given to the *Content Operations Team* for processing.

The video comes in a high definition format that's many terabytes in size. A terabyte is big. Imagine 60 stacks of paper as tall as the Eiffel Tower. That's a terabyte.

Before you can view a video, Netflix puts it through a rigorous multi-step process.



Netflix

Validating the video.

The first thing Netflix does is spend a lot of time validating the video. It looks for digital artifacts, color changes, or missing frames that may have been caused by previous transcoding attempts or data transmission problems.

The video is rejected if any problems are found.

Into the media pipeline.

After the video is validated, it's fed into what Netflix calls the *media pipeline*.

A *pipeline* is simply a series of steps data is put through to make it ready for use, much like an assembly line in a factory. More than 70 different pieces of software have a hand in creating every video.

It's not practical to process a single multi-terabyte sized file, so the first step of the pipeline is to break the video into lots of smaller chunks.

The video chunks are then put through the pipeline so they can be encoded *in parallel*. In parallel simply means the chunks are processed at the same time.

Let's illustrate parallelism with an example.



Let's say you have one hundred dirty dogs that need washing. Which would be faster, one person washing the dogs one after another? Or would it be faster to hire one hundred dog washers and wash them all the same time?

Obviously, it's faster to have one hundred dog washers working at the same time. That's parallelism. And that's why Netflix uses so many servers in EC2. They need a lot of servers to process these huge video files in parallel. It works too. Netflix says a source media file can be encoded and pushed to their CDN in as little as 30 minutes.

Once the chunks are encoded, they're validated to make sure no new problems have been introduced.

Then the chunks are assembled back into a file and validated once again.

The result is a pile of files.

The encoding process creates a lot of files. Why? The end goal for Netflix is to support every internet-connected device.

Netflix started streaming video in 2007 on Microsoft Windows. Over time more devices were added—Roku, LG, Samsung Blu-ray, Apple Mac, Xbox 360, LG DTV, Sony PS3, Nintendo Wii, Apple iPad, Apple iPhone, Apple TV, Android, Kindle Fire, and Comcast X1.

In all, Netflix supports 2200 different devices. Each device has a video

format that looks best on that particular device. If you're watching Netflix on an iPhone, you'll see a video that gives you the best viewing experience on the iPhone.

Netflix calls all the different formats for a video its *encoding profile*.

Netflix also creates files optimized for different network speeds. If you're watching on a fast network, you'll see higher quality video than you would if you're watching over a slow network.

There are also files for different audio formats. Audio is encoded into different levels of quality and in different languages.

There are also files included for subtitles. A video may have subtitles in a number of different languages.

There are a lot of different viewing options for every video. What you see depends on your device, your network quality, your Netflix plan, and your language choice.

Just how many files are we talking about?

For *The Crown*, Netflix stores around 1,200 files!

Stranger Things season 2 has even more files. It was shot in 8K and has nine episodes. The source video files were many, many terabytes of data. It took 190,000 CPU hours to encode just one season.

The result? 9,570 different video, audio, and text files!

Let's see how Netflix plays all that video.

Three Different Strategies for Streaming Video

Netflix has tried three different video streaming strategies its own small CDN; third-party CDNs; and Open Connect.

Let's start by defining CDN. A CDN is a *content distribution network*.

Content for Netflix—is of course—the video files we discussed in the previous section.

Distribution means video files are copied from a central location, over a network and stored on computers located all over the world.

For Netflix, the central location where videos are stored is S3.

Why build a CDN?

The idea behind a CDN is simple: put video as close as possible to users by spreading computers throughout the world. When a user wants to watch a video, find the nearest computer with the video on it and stream to the device from there.

The biggest benefits of a CDN are speed and reliability.

Imagine you're watching a video in London and the video is being streamed from Portland, Oregon. The video stream must pass through a lot of networks, including an undersea cable, so the connection will be slow and unreliable.

By moving video content as close as possible to the people watching it, the viewing experience will be as fast and reliable as possible.

Each location with a computer storing video content is called a PoP or *point of presence*. Each PoP is a physical location that provides access to the internet. It houses servers, routers, and other telecommunications equipment. We'll talk more about PoPs later.

The First CDN Was Too Small

In 2007, when Netflix debuted its new streaming service, it had 36 million members in 50 countries, watching more than a billion hours of video each month, streaming multiple terabits of content per second.

To support the streaming service, Netflix built its own simple CDN in five different locations within the United States.

The Netflix video catalog was small enough at the time that each location contained all of its content.

The Second CDNs Were Too Big

In 2009, Netflix decided to use 3rd-party CDNs. Around this time, the pricing for 3rd-party CDNs was coming down.

Using 3rd-party CDNs made perfect sense for Netflix. Why spend all the time and effort building a CDN of your own when you can instantly reach the globe using existing CDN services?

Netflix contracted with companies like Akamai, Limelight, and Level 3 to provide CDN services. There's nothing wrong with using third-party CDNs. In fact, pretty much every company does. For example, the NFL has used Akamai to stream live football games.

By not building out its own CDN, Netflix had more time to work on other higher priority projects.

Netflix put a lot of time and effort into developing smarter clients. Netflix created algorithms to adapt to changing networks conditions. Even in the face of errors, overloaded networks, and overloaded servers, Netflix wants members always viewing the best picture possible. One technique Netflix developed is switching to a different video source—say another CDN, or a different server—to get a better result.

At the same time, Netflix was also devoting a lot of effort into all the AWS services we talked about earlier. Netflix calls the services in AWS its *control plane*. Control plane is a telecommunications term identifying the part of the system that controls everything else. In your body, your brain is the control plane; it controls everything else.

Then Netflix thought it could do better by developing its own CDN.

Open Connect Was Just Right

In 2011, Netflix realized at its scale it needed a dedicated CDN solution to maximize network efficiency. Video distribution is a core competency for Netflix and could be a huge competitive advantage.

So Netflix started developing Open Connect, its own purpose-built CDN.

Open Connect launched in 2012.

Open Connect has a lot of advantages for Netflix:

- *Less expensive.* 3rd-party CDNs are expensive. Doing it themselves would save a lot of money.
- *Better quality.* By controlling the entire video path—transcoding, CDN, clients on devices—Netflix reasoned it could deliver a superior video viewing experience.
- *More Scalable.* Netflix has the goal of providing service everywhere in the world. Quickly supporting all those people while providing a quality video viewing experience required building its own system.

The 3rd-party CDNs must support users accessing any kind of content from anywhere in the world. Netflix has a much simpler job.

Netflix knows exactly who its users are because they must subscribe to Netflix. Netflix knows exactly which videos it needs to serve. Just knowing it only has to serve large video streams allows Netflix to make a lot of smart optimization choices other CDNs can't make. Netflix also knows a lot about its members. The company knows which videos they like to watch and when they like to watch them.

With this kind of knowledge, Netflix built a really high-performing CDN. Let's go into more details on how Open Connect works.

Open Connect Appliances

Remember how we said a CDN has computers distributed all over the world?

Netflix developed its own computer system for video storage. Netflix calls them Open Connect Appliances or OCAs.

Here's what an early OCA installation in a site looked like:



Netflix

There are many OCAs in the above picture. OCAs are grouped into clusters of multiple servers.

Each OCA is a fast server, highly optimized for delivering large files, with lots and lots of hard disks or flash drives for storing video.

Here's what one of the OCA servers looks like:

Open Connect Appliance - Global



Netflix

There are several different kinds of OCAs for different purposes. There are large OCAs that can store Netflix's entire video catalog. There are smaller OCAs that can store only a portion of Netflix's video catalog. Smaller OCAs are filled with video every day, during off-peak hours, using a process Netflix calls *proactive caching*. We'll talk more about how proactive caching works later.

From a hardware perspective, there's nothing special about OCAs. They're based on commodity PC components and assembled in custom cases by various suppliers. You could buy the same computers if you wanted to.

Notice how all Netflix's computers are red? Netflix had their computers specially made to match their logo color.

From a software perspective, OCAs use the FreeBSD operating system and NGINX for the web server. Yes, every OCA has a web server. Video streams using NGINX. If none of these names make any sense, don't worry, I'm just including them for completeness.

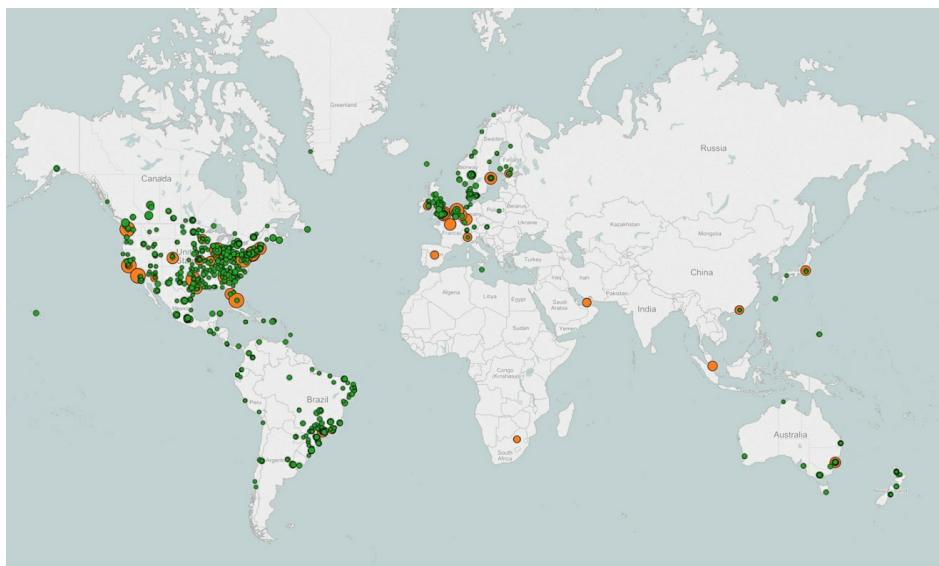
The number of OCAs on a site depends on how reliable Netflix wants the site to be, the amount of Netflix traffic (bandwidth) that is delivered from that site, and the percentage of traffic a site allows to be streamed.

When you press play, you're watching video streaming from a specific OCA, like the one above, in a location near you.

For the best possible video viewing experience, what Netflix would really like to do is cache video in your house. But that's not practical yet. The next best thing is to put a mini-Netflix as close to your house as they can. How do they do that?

Where does Netflix put Open Connect Appliances (OCAs)?

Netflix delivers huge amounts of video traffic from thousands of servers in more than 1,000 locations around the world. Take a look at this map of video serving locations:



Netflix

Other video services, like YouTube and Amazon, deliver video on their own backbone network. These companies literally built their own global network for delivering video to users. That's very complicated and very expensive to do.

Netflix took a completely different approach to building its CDN.

Netflix doesn't operate its own network; it doesn't operate its own datacenters anymore either. Instead, internet service providers (ISPs) agree to

put OCAs in their datacenters. OCAs are offered free to ISPs to embed in their networks. Netflix also puts OCAs in or close to internet exchange locations (IXPs).

Using this strategy Netflix doesn't need to operate its own datacenters, yet it gets all the benefits of being in a regular datacenter it's just someone else's datacenter. Genius!

Those last two paragraphs were pretty dense, so let's break it down.

Using ISPs to build a CDN.

An ISP is your internet provider. It's who you get your internet service from. It might be Verizon, Comcast, or thousands of other services.

The main point here is ISPs are located all around the world and they're close to customers. By placing OCAs in ISP datacenters, Netflix is also all over the world, and close to its customers.

Using IXPs to build a CDN.

An internet exchange location is a datacenter where ISPs and CDNs exchange internet traffic between their networks. It's just like going to a party to exchange Christmas presents with your friends. It's easier to exchange presents if everyone is in one place. It's easier to exchange network traffic if everyone is one place.

IXPs are located all over the world:



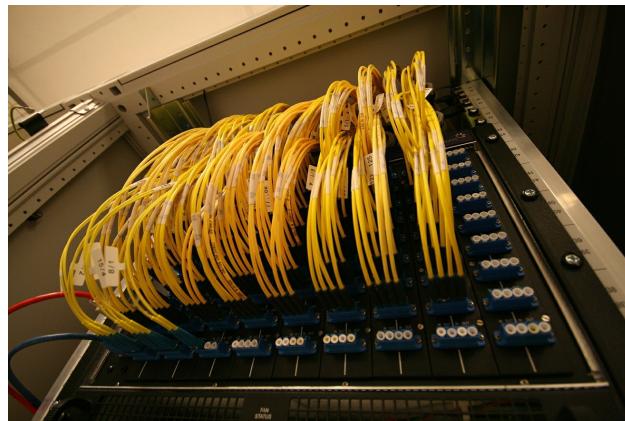
TeleGeography's Internet Exchange Map

Here's what the London Internet Exchange looks like:



London Internet Exchange (LINX)

Drill down on those yellow fiber optic cables and what you'll see is something like this from the AMS-IX Internet exchange point, in Amsterdam, Netherlands:



Wikimedia Commons

Each wire in the above picture connects one network to another network. That's how different networks exchange traffic with each other.

An IXP is like a highway interchange, only using wires:



Wikimedia Commons

For Netflix, this is another win. IXPs are all over the world. So by putting their OCAs in IXPs, Netflix doesn't have to run its own datacenters.

Video is Proactively Cached to OCAs Every Day

Netflix has all this video sitting in S3. They have all these video serving computers spread throughout the world. There's just one thing missing: Video!

Netflix uses a process it calls *proactive caching* to efficiently copy video to OCAs.

What is a cache?

A cache is a hiding place, especially one in the ground, for ammunition, food and treasures.

You know how squirrels bury nuts for the winter?



Each location they bury nuts is a *cache*. During the winter, any squirrel can find a nut cache and chow down.

Arctic explorers sent small teams ahead to cache food, fuel and other supplies along the route they were taking. The larger team following behind would stop at every cache location and resupply.

Both the squirrels and Arctic explorers were being *proactive*; they were doing something ahead of time to prepare for later.

Each OCA is a video cache of what you'll most likely want to watch.

Netflix caches video by predicting what you'll want to watch.

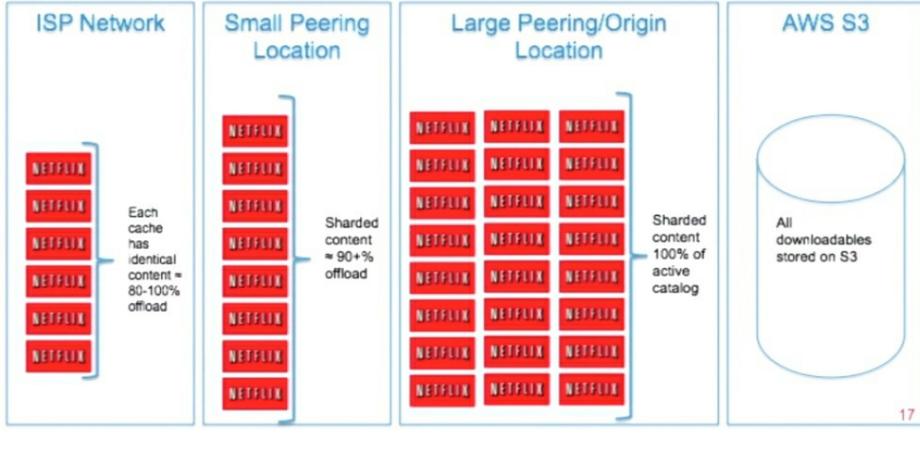
Everywhere in the world, Netflix, knows to a high degree of accuracy what its members like to watch and when they like to watch it. Remember how we said Netflix was a data-driven company?

Netflix uses its popularity data to *predict* which videos members probably will want to watch tomorrow in each location. Here, *location* means a cluster of OCAs housed within an ISP or IXP.

Netflix copies the predicted videos to one or more OCAs at each location. This is called *prepositioning*. Video is placed on OCAs before anyone even asks.

This gives great service to members. The video they want to watch is already close to them, ready and available for streaming.

Netflix operates what is called a *tiered caching system*.



Netflix

The smaller OCAs we talked about earlier are placed in ISPs and IXPs. These are too small to contain the entire Netflix catalog of videos. Other locations have OCAs containing most of Netflix's video catalog. Still, other locations have big OCAs containing the entire Netflix catalog. These get their videos from S3.

Every night, each OCA wakes up and asks a service in AWS which videos it should have. The service in AWS sends the OCA a list of videos it's supposed to have based on the predictions we talked about earlier.

Each OCA is in charge of making sure it has all the videos on its list. If an OCA in the same location has one of the videos it's supposed to have, then it will copy the video from the local OCA. Otherwise, a nearby OCA with the video will be found and copied.

Since Netflix forecasts what will be popular tomorrow, there's always a one day lead time before a video is required to be on an OCA. This means videos can be copied during quiet, off-peak hours, substantially reducing bandwidth usage for ISPs.

There's never a *cache miss* in Open Connect. A cache miss would be asking for a specific video from an OCA and the OCA saying it doesn't have it. Cache misses happen all the time on other CDNs because you can't afford to copy content everywhere. Since Netflix knows all the videos it must cache, it knows exactly where each video is at all times. If a smaller OCA doesn't have a video, then one of the larger OCAs is always guaranteed to have it.

Why doesn't Netflix just copy all their video to every OCA in the world? Its video catalog is way too large to store everything at all locations. In 2013, the video catalog for Netflix was over 3 petabytes; I have no idea how large it is today, but I can only assume it's significantly larger.

That's why Netflix developed the method of choosing which videos to store on each OCA using data to *predict* what their members will want to watch.

Let's take an example. *House of Cards* is a very popular show. Which OCAs should it be copied to? Probably every location because members worldwide will want to watch *House of Cards*.

What if a video isn't as popular as *House of Cards*? Netflix decides which locations it should be copied to in order to best serve nearby member requests.

Within a location, a popular video like *House of Cards* is copied to many different OCAs. The more popular a video, the more servers it will be copied to. Why? If there was only one copy of a very popular video, streaming the video to members would overwhelm the server. As they say, many hands make light work.

A video isn't considered live when it's copied to just one OCA. Netflix wants to be able to play the same content at the same time everywhere in the world. Only when there are a sufficient number of OCAs with enough copies of the video to serve it appropriately, will the video be considered live and ready for members to watch.

Daredevil Season 2 in 2016, for example, was the first time Netflix released all episodes of a show, on all devices, in all countries, at the same time.

Hosting OCAs: What's in it for ISPs?

Why would an ISP agree to put an OCA cluster inside their network? At first blush, it seems too generous, but you'll be happy to know it's rooted firmly in self-interest.

To understand why, we'll need to talk about how networks work. I know throughout this book we've said cloud services are accessed over the internet.

That's not the case for Netflix, at least when watching a video. When using a Netflix app, it talks to AWS over the internet.

The internet is an interconnect of networks. You have an ISP that provides internet service. I get my internet service from Comcast. What that means is my house connects to Comcast's network using a fiber optic cable. Comcast's network is their network; it's not the internet, the internet is something else.

Let's say I want to do a Google search, and I type a query into my browser and hit enter.

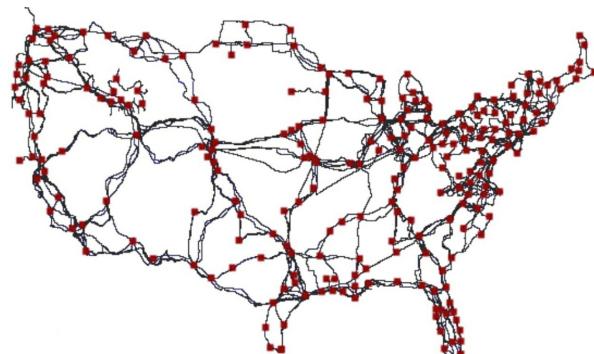
My request to Google first flows over Comcast's network. Google isn't on Comcast's network. At some point, my request has to go to Google's network. That's what the internet is for.

The internet connects Comcast's network to Google's network. There are these things called *routing protocols* that act like a traffic cop, directing where network traffic goes.

When my Google query is routed onto the internet it's not on Comcast's network anymore, and it's not on Google's network. It's on what's called the *internet backbone*.

The internet is woven together from many privately owned networks that choose to interoperate with each other. The IXPs we looked at earlier are one way networks connect with each other.

In the United States, here's a map of the long haul fiber network:



InterTubes: A Study of the US Long-haul Fiber-optic Infrastructure

What Netflix has done with Open Connect is placed its OCA clusters inside the ISPs network. That means if I watch a Netflix video I'll be talking to an OCA in Comcast's network. All my video traffic is on Comcast's network; it never hits the internet.

The key to scaling video delivery is to be as close to users as possible. When you're doing that you're not using the internet backbone. Requests are being satisfied on a local part of the network.

Why is this a good thing? Recall that we said Netflix already consumes more than 37% of the internet traffic in the United States. If ISPs didn't cooperate, Netflix would use even more of the internet. The internet couldn't handle all the video traffic. ISPs would have to add a lot more network capacity, and that's expensive to build.

Right now, up to 100% of Netflix content is being served from within ISP networks. This reduces costs by relieving internet congestion for ISPs. At the same time, Netflix members experience a high-quality viewing experience. And network performance improves for everyone.

It's a win-win.

Open Connect is Reliable and Resilient

Earlier we discussed how Netflix increased the reliability of its system by running out of three different AWS regions. The architecture of Open Connect accomplished the same goal.

What may not be immediately obvious is that the OCAs are independent of each other. OCAs act as self-sufficient video-serving archipelagos. Members streaming from one OCA are not affected when other OCAs fail.

What happens when an OCA fails? The Netflix client you're using immediately switches to another OCA and resumes streaming.

What happens if too many people in one location use an OCA? The Netflix client will find a more lightly loaded OCA to use.

What happens if the network a member is using to stream video becomes overloaded? The same sort of thing. The Netflix client will find another OCA

on a better performing network.

Open Connect is a very reliable and resilient system.

Netflix Controls the Client

Netflix handles failures gracefully because it controls the client on every device running Netflix.

Netflix develops its Android and iOS apps themselves, so you might expect them to control those. But even on platforms like Smart TVs, where Netflix doesn't build the client, Netflix still has control because it controls the *software development kit* (SDK).

A SDK is *a set of software development tools that allows the creation of applications*. Every Netflix app makes requests to AWS and plays video using the SDK.

By controlling the SDK, Netflix can adapt consistently and transparently to slow networks, failed OCAs, and any other problems that might arise.

Finally: Here's What Happens when You Press Play

It's been a long road, but we're finally here. We're ready to press play and watch a movie!

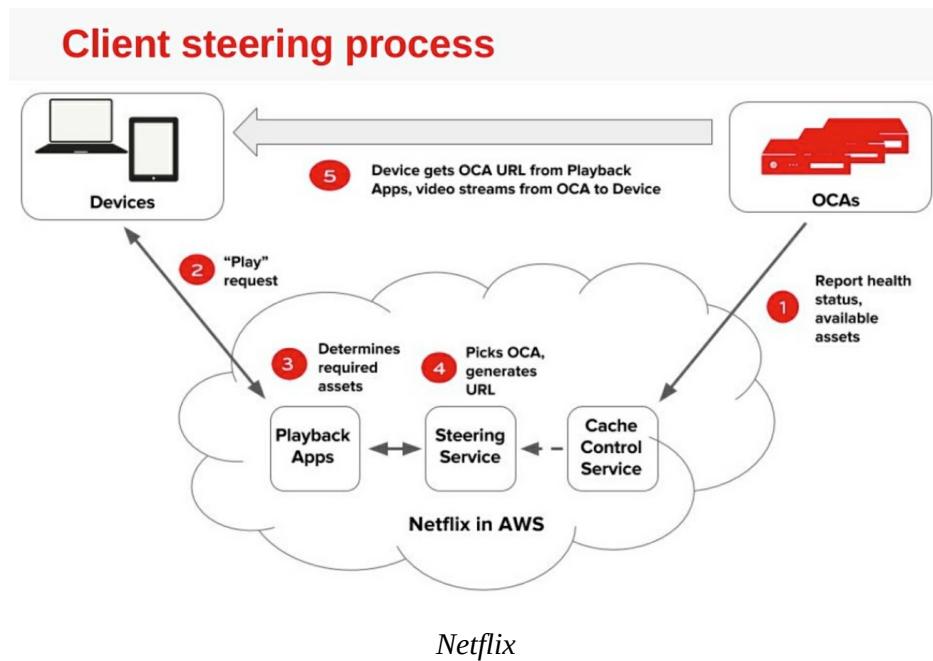


Netflix

We've learned a lot. Here's what we've learned so far:

- Netflix can be divided into three parts: the backend, the client, and the CDN.
- All requests from Netflix clients are handled in AWS.
- All video is streamed from a nearby Open Connect Appliance (OCA) in the Open Connect CDN.
- Netflix operates out of three AWS regions and can usually handle a failure in any region without members even noticing.
- New video content is transformed by Netflix into many different formats so the best format can be selected for viewing based on the device type, network quality, geographic location, and the member's subscription plan.
- Every day, over Open Connect, Netflix distributes video throughout the world, based on what they predict members in each location will want to watch.

Here's a picture of how Netflix describes the play process:



Now, let's complete the picture:

- You select a video to watch using a client running on some device. The client sends a *play* request, indicating which video you want to

play, to Netflix's *Playback Apps* service running in AWS.

- We've not talked about this before, but a big part of what happens after you hit play has to do with licensing. Not every location in the world has a license to view every video. Netflix must determine if you have a valid license to view a particular video. We won't talk about how that works—it's really boring—but keep in mind it's always happening. One reason Netflix started developing its own content is to avoid licensing issues. Netflix wants to release a show to everyone in the world all at the same time. Creating its own content is the easiest way for Netflix to avoid worrying about licensing problems.
- Taking into account all the relevant information, the Playback Apps service returns URLs for up to ten different OCA servers. These are the same sort of URLs you use all the time in your web browser. Netflix uses your IP address and information from ISPs to identify which OCA clusters are best for you to use.
- The client intelligently selects which OCA to use. It does this by testing the quality of the network connection to each OCA. It will connect to the fastest, most reliable OCA first. The client keeps running these tests throughout the video streaming process.
- The client probes to figure out the best way to receive content from the OCA.
- The client connects to the OCA and starts streaming video to your device.
- Have you noticed when watching a video the picture quality varies? Sometimes it will look pixelated, and after awhile the picture snaps back to HD quality? That's because the client is adapting to the quality of the network. If the network quality declines, the client lowers video quality to match. The client will switch to another OCA when the quality declines too much.

That's what happens when you press play on Netflix. Who would have ever thought so simple a thing as watching a video was so complex?