A 360 Degree View Of The Entire Netflix Stack

Application & Data

Languages	Java, Python, Javascript
Database	MySQL, Cassandra, Oracle
Frameworks	Node.js
Cloud Hosting	Amazon EC2
Javascript UI Library	React
SQL Database-as-a-Service	Amazon RDS
NoSQL Database-as-a-Service	Amazon DynamoDB
Database Cluster Management	Dynomite

Business Tools

Productivity Suite	Google Apps
Project Management	Confluence
Password Management	OneLogin

Utilities

Transactional Email	Amazon SES
Mobile Push Messaging	Urban Airship
API Tools	Falcor

DevOps

Code Collaboration & Version Control	GitHub
Continuous Integration	Jenkins
Server Management	Apache Mesos
Log Management	Sumo Logic
Mobile Error Monitoring	Crittercism
Performance Monitoring	Boundary, LogicMonitor

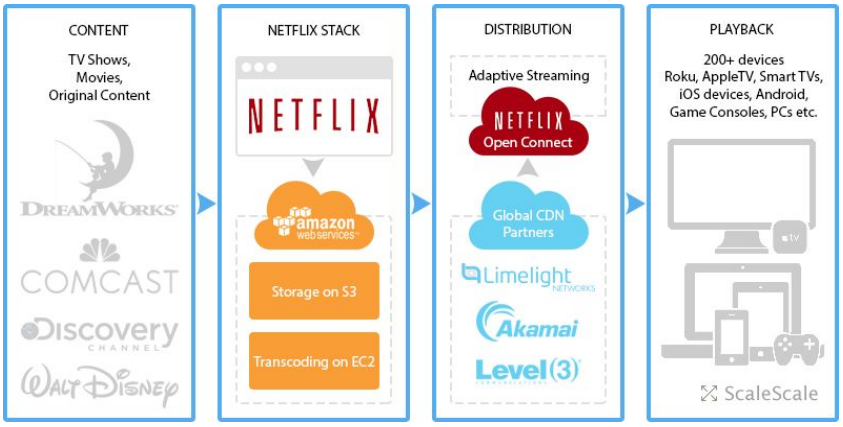
Open Connect CDN

Operating System	FreeBSD
Server	Nginx
Routing	Bird daemon

Supporting Many Titles With Amazon

Netflix's infrastructure is on Amazon EC2 with master copies of digital films from movie studios being stored on Amazon S3. Each film is encoded into over 50 different versions based on video resolution and audio quality using machines on the cloud. Over 1 petabyte of data is stored on Amazon. These data are sent to content delivery networks to feed the content to local ISPs.

Netflix uses a number of open-source software at the backend, including Java, MySQL, Gluster, Apache Tomcat, Hive, Chukwa, Cassandra, and Hadoop.



File Supported

- **Video – VC-1, H.264 (AVC), VC-1, H.263, H.265 (HEVC)**
- **Audio – WMA, Dolby Digital, Dolby Digital Plus, AAC and Ogg Vorbis**

Netflix Open Connect CDN

The Netflix Open Connect CDN is provided for larger ISPs that have over 100,000 subscribers. A specially built low power high storage density appliance caches Netflix content within the ISPs' data centers to reduce internet transit costs. This appliance runs the FreeBSD operating system, nginx and the Bird Internet routing daemon.

NetFlix Paris Open Connect – Photo Credit: @dtemkin twitter

Watch the Open Connect video here.

Scaling Algorithms

In 2009, Netflix did a contest called the Netflix prize. They opened up a bunch of anonymized data and allowed teams to try and derive better algorithms. They got a 10.06% uplift of their existing algorithm from the winning team. Netflix was going to run another Netflix Prize but ultimately didn't because of privacy concerns from the FTC.

The Netflix recommendation system consists of many algorithms. The two core algorithms used in their production system are Restricted Boltzmann Machines (RBM) and a form of Matrix Factorization called SVD++. These two algorithms are combined using a linear blend to produce a single higher accuracy estimate.

Restricted Boltzmann Machines are neural networks that have been modified to work in collaborative filtering. Each user has one RBM with the input node for each representing a movie the user has rated.

SVD++ is an asymmetric form of SVD (Singular Value Decomposition) that makes use of implicit information like RBMs. It was developed by the winning team in the Netflix Prize contest.

On their Engineering blog, the Netflix team covers Learning a Personalized Homepage

1. Restricted Boltzmann Machines (RBM)
2. Matrix Factorization called SVD++

Open Source Projects

https://netflix.github.io/. Netflix has a great engineering blog and they recently did a post called The Evolution of Open Source at Netflix.

Big Data

- Genie - A powerful, REST-based abstraction to our various data processing frameworks, notably Hadoop.
- Inviso - provides detailed insights into the performance of our Hadoop jobs and clusters.
- Lipstick - Shows the workflow of Pig jobs in a clear, visual fashion.
- Aegisthus - Enables the bulk abstraction of data out of Cassandra for downstream analytic processing.

Build And Delivery Tools

- Nebula - Effort at Netflix to share its internal build infrastructure.
- Aminator - A tool for creating EBS AMIs.
- Asgard - Web interface for application deployments and cloud management in Amazon Web Services (AWS).

Common Runtime Services & Libraries

- Eureka - Service discovery for the Netflix cloud platform.
- Archaius - Distributed configuration.
- Ribbon - Resilent and intelligent inter-process and service communication.
- Hystrix - Provides reliability beyond single service calls. Isolates latency and fault tolerance at runtime.
- Karyon and Governator - JVM container services.
- Prana sidecar - Prana provides proxy capabilities within an instance.
- Zuul - Provides dyamically scriptable proxying at the edge of the cloud deployment.
- Fenzo - Provides advanced scheduling and resource management for cloud native frameworks.

Data Persistence

- EVCache and Dynomite - For using Memcached and Redis at scale.
- Astyanax and Dyno - Client libraries to better consume datastores in the Cloud.

Insight, Reliability And Performance

- Atlas - Time-series telemetry platform
- Edda - Service to track changes in your cloud
- Spectator - Easy integration of Java application code with Atlas
- Vector - Exposes high-resolution host-level metrics with minimal overhead.
- Ice - Exposes ongoing cost and and cloud utilization trends.
- Simian Army - Tests Netflix instances for random failures.

Security

- Security Monkey - Helps monitor and secure large AWS-based environments.
- Scumblr - Leverages Internet-wide targeted searches to surface specific security issues for investigation.
- MSL - An extensible and flexible secure messaging protocol that addresses a number of secure communications use cases and requirements.
- Falcor - Represent remote data sources as a single domain model via a virtual JSON graph.
- Restify - node.js REST framework specifically meant for web service APIs
- RxJS - A reactive programming library for JavaScript

References

1. On HackerNews
2. https://en.wikipedia.org/wiki/Netflix
3. http://gizmodo.com/this-box-can-hold-an-entire-netflix-1592590450
4. http://edition.cnn.com/2014/07/21/showbiz/gallery/netflix-history/
5. http://techblog.netflix.com/2015/01/netflixs-viewing-data-how-we-know-where.html
6. https://gigaom.com/2013/03/28/3-shades-of-latency-how-netflix-built-a-data-architecture-around-timeliness/
7. https://gigaom.com/2015/01/27/netflix-is-revamping-its-data-architecture-for-streaming-movies/
8. http://stackshare.io/netflix/netflix
9. https://www.quora.com/How-does-the-Netflix-movie-recommendation-algorithm-work
10. https://netflix.github.io/