

System Design Day 12: Video Stream Sytem

Video Streaming System Design – Definition & Example

Definition

A **video streaming system** is a distributed platform that delivers video content to users in real time over the internet without requiring a full download. It uses **adaptive bitrate streaming**, **content delivery networks (CDNs)**, and **media encoding pipelines** to ensure smooth playback across devices and network conditions, while enforcing **digital rights management (DRM)** and providing scalability for millions of concurrent viewers.

1. Requirements

Functional:

- User authentication & subscription validation.
- Browsing/search of content library.
- Adaptive streaming based on bandwidth.
- Resume-watch & multi-device sync.
- DRM for licensed content.

Non-Functional:

- Low startup latency (<2 seconds).
 - High throughput for global audiences.
 - 99.99% availability.
 - Scalability & fault tolerance across data centers.
-

2. High-Level Architecture

Example: Netflix / Amazon Prime

1. **Client Devices** – Smart TVs, mobiles, browsers.
2. **API Gateway** – Routes requests to backend microservices.
3. **User Services** – Authentication, profiles, subscription checks.
4. **Content Catalog Service** – Metadata, recommendations, search.
5. **CDN** – Stores and delivers video chunks close to the user.
 - Netflix → **Open Connect** (custom CDN).
 - Amazon Prime → **AWS CloudFront**.
6. **Storage & Transcoding Pipelines** –
 - Raw media stored in S3/Blob Storage.
 - Transcoded into multiple bitrates/resolutions for adaptive streaming.
7. **DRM Services** – Widevine, FairPlay, PlayReady.
8. **Streaming Protocols** – HLS (HTTP Live Streaming) / MPEG-DASH.
9. **Monitoring & Analytics** – Playback quality, network metrics, A/B tests.

Flow Example (Netflix Movie Playback)

User selects a movie → API Gateway verifies subscription → Returns playback token + nearest CDN URL → Client streams via HLS from local CDN node → Adaptive bitrate adjusts dynamically to network conditions.

3. Key Trade-offs

Design Choice	Pros	Cons
Custom CDN (Netflix) vs 3rd-Party CDN (Amazon Prime)	Optimized routing, full control, reduced long-term cost	High upfront & maintenance cost
Adaptive Bitrate Streaming (ABR)	Smooth playback on varying bandwidth	More storage needed for multiple versions

Design Choice	Pros	Cons
Pre-fetching / Edge Caching	Reduces startup delay	Higher storage costs on edge nodes
Microservices Architecture	Independent scaling, fault isolation	Inter-service latency, operational complexity
DRM Enforcement	Protects intellectual property	Adds encryption/decryption overhead
Global Content Replication	Low latency worldwide	High replication cost

4. Example in Numbers

- **Netflix:** 270+ million subscribers, 100B+ hours streamed yearly.
- **Open Connect** appliances in 100+ countries.
- Bitrate switching within ~200ms for smooth playback.