Content Delivery Networks (CDN): Caching Principles, Architecture, and Resource Optimization

Scott Wainner, Distinguished Systems Engineer
Questions?
Use Cisco Spark to communicate with the speaker after the session

How
1. Find this session in the Cisco Live Mobile App
2. Click “Join the Discussion”
3. Install Spark or go directly to the space
4. Enter messages/questions in the space

cs.co/ciscolivebot#BRKSPV-2160
Objectives

• Understand Principles of Caching
• Understand System Architecture and Role of CDN
• Distinguish Impact of Service Delivery Models
• Assess Characteristics of Resource Optimization
Agenda

• Caching Principles
• Caching Value Proposition Models
• Content Delivery Functions
• Content Delivery of Media Services
• Platform Optimization and Management
• Enabling Cloud Content Delivery
Content Caching Principles
Relevance
Many Media Types and Purposes

• Commercial Video
• Imagery
• Production Training / Education Systems
• Marketing / Mass Communication Systems
• Notification Information Distribution
• Telemetry Distribution to Subscribers
• Software Distribution
Content Distribution Principles

Efficiency

Performance

Resiliency
Content Distribution Architectural Models

- Hierarchical
  - Distribution Tree from Origin
  - Often associated with an Authoritative Source
  - Tightly controlled distribution policies

- Peer to peer
  - Distributed Hash Table model
  - Content can be cached anywhere
  - Appropriate in fully meshed topologies
  - Multiple sources
System Architecture

Content Management
- Product Bundles
- Workflows
- Catalog Creation
- Entitlement
- Policy
- Digital Rights

Offer Management
- Personalization
- Billing
- Offers
- Subscriptions

Origin Servers
- Live Origin Servers
- VOD Origin Servers

Content Delivery
- Live Media
- VOD Media

Content Distribution
- Live ABR traffic
- VOD ABR traffic

Clients
- Authentication
- Navigation
- Entitlement & Session Control
- Credentials

Video Control Plane
- Live EPG
- VoD IMDb
- Image Tags
- Asset Management

Data Plane
- Transcoding
- Live
- On Demand
- Streams
- Images
- Software Video

Content Acquisition
- Authorization & Assignment

Our Focus Today
- Metadata
- Video Control

© 2018 Cisco and/or its affiliates. All rights reserved. Cisco Public
CDN Systems Architecture

- Centralized Content, Command and Control, and Analytics
- Hierarchy of Origination, Mid-Tier and Edge Caching
- Management Plane, Content Routing Plane, Content Delivery Plane
- Resiliency Defined at each Tier
- Cost Optimized Platform Selection

Hierarchy of Origination, Mid-Tier and Edge Caching

- Ingest, Routing, & Management (Optional) Caching Layer
- Optimized Edge Cache

Resiliency Defined at each Tier

Cost Optimized Platform Selection

Centralized Content, Command and Control, and Analytics

Ingest, Routing, & Management

Internet Content

Published Content

Programming

Internet Content

Published Content

Programming

Content Library (Origin)

Traffic Servers

Traffic Ops

Traffic Routing

Content Routing Plane, Content Delivery Plane

Centralized Content, Command and Control, and Analytics

Hierarchy of Origination, Mid-Tier and Edge Caching

Management Plane, Content Routing Plane, Content Delivery Plane

Resiliency Defined at each Tier

Cost Optimized Platform Selection

Centralized Content, Command and Control, and Analytics

Hierarchy of Origination, Mid-Tier and Edge Caching

Management Plane, Content Routing Plane, Content Delivery Plane

Resiliency Defined at each Tier

Cost Optimized Platform Selection

Ingest, Routing, & Management (Optional) Caching Layer

Optimized Edge Cache

Ingest, Routing, & Management

Internet Content

Published Content

Programming

Content Library (Origin)

Traffic Servers

Traffic Ops

Traffic Routing

Internet Content

Published Content

Programming

Content Library (Origin)

Traffic Servers

Traffic Ops

Traffic Routing

Ingest, Routing, & Management (Optional) Caching Layer

Optimized Edge Cache

Centralized Content, Command and Control, and Analytics

Hierarchy of Origination, Mid-Tier and Edge Caching

Management Plane, Content Routing Plane, Content Delivery Plane

Resiliency Defined at each Tier

Cost Optimized Platform Selection

Ingest, Routing, & Management

Internet Content

Published Content

Programming

Content Library (Origin)

Traffic Servers

Traffic Ops

Traffic Routing

Ingest, Routing, & Management (Optional) Caching Layer

Optimized Edge Cache

Centralized Content, Command and Control, and Analytics

Hierarchy of Origination, Mid-Tier and Edge Caching

Management Plane, Content Routing Plane, Content Delivery Plane

Resiliency Defined at each Tier

Cost Optimized Platform Selection

Ingest, Routing, & Management

Internet Content

Published Content

Programming

Content Library (Origin)

Traffic Servers

Traffic Ops

Traffic Routing

Ingest, Routing, & Management (Optional) Caching Layer

Optimized Edge Cache

Centralized Content, Command and Control, and Analytics

Hierarchy of Origination, Mid-Tier and Edge Caching

Management Plane, Content Routing Plane, Content Delivery Plane

Resiliency Defined at each Tier

Cost Optimized Platform Selection
Content Caching Techniques
CDN - Introduction to Dynamic Caching

• Concepts
  • CDN is a “Proxy” for Origin Servers

• Redirecting clients to CDN

• CDN Functional Cache Elements
  • “Traffic Routing” Redirection
  • “Origin Server” Library
  • “Traffic Server” Caching
  • “Traffic Server” Edge Cache
CDN – Implementation Considerations

- Storage Considerations
  - Live Media (Finite Window)
  - VoD Media (Natural Expiration)
  - Cloud DVR (No Caching)

- Content Affinity
  - Localization
  - Mobile
  - Pre-positioned

- Mapping Methodology
  - File Handle in URL
  - Byte Offset in URL

- Retention / Eviction Algorithms
  - Least Recently Used
  - First In / First Out
  - Metadata (Algorithmic Expiration)
HTTP Caching

- RFC2616 HTTP/1.1
  - RFC 6585 Status Codes
  - RFC 2817 TLS
  - RFC 5785 Well-Known URI
  - RFC 6266 Content-Disposition

- Cache Types
  - Transparent Caches
  - Proxy Caches
Transparent Caching

- Unauthorized Intercept
  - Non Monetized

- Dynamic Interpretation of Request
  - HTTP GET Analysis

- Authentication or Encryption Interference
  - Authentication
    - Token or HASH Validation Corrupted

- Payload Encryption
  - (e.g. Media Headers, Opaque Objects)

- Transport Encryption
  - (e.g. TLS, SPDR)
HTTP Caching

- RFC2616 HTTP/1.1
  - RFC 6585 Status Codes
  - RFC 2817 TLS
  - RFC 5785 Well-Known URI
  - RFC 6266 Content-Disposition

- Cache Types
  - Transparent Caches
  - Proxy Caches
Proxy Caching

- Authorized Delegation
- Explicit Interpretation Provided to Cache
  - Optimized Payload is Known a priori
- Authentication or Encryption Viable
  - Authenticators Validated and Recalculated
    - Tokens, HASH
    - Private Keys Shared
  - Payload Remains Encrypted
    - Substitution of Opaque Payload
  - Transport Encryption
    - Public Certificates Exchanged
CDN Caching Basics

- Authorized Delegation
- Explicit Interpretation Provided to Cache
- Authentication or Encryption Viable
- Redirection to Optimal Location
- Cache Hit Ratio
  - Distributed Edge
    - Edge Cache
  - Intermediate Layer
    - Reverse Proxy Cache
CDN Value Proposition
Content Caching Principles

**Graph:**
- **Cost vs. Caching Sites:**
  - Bandwidth Costs
  - Cache Costs
  - Optimized Costs

**Graph:**
- **Cache Hit Rates vs. Cache size as % of dataset:**
- **Cache Hit Rates:**
  - 0% to 100%

**Content Popularity:**
- **Top 100 Titles:**
  - More views
- **Bottom 100 Titles:**
  - Fewer views
Content Popularity

Content Popularity Often Characterized by Zipf’s Law

Library Described by $\alpha$ (Alpha)

Each Library is Unique
- Old Movies
- Current Hits
- Music Videos
- Sports Highlights
- TV Series

Views

Top 100

Cache

Titles ranked by Demand

Bottom 100
Contributing Factors

• CHR of Content Types
  • ABR more difficult to predict than for PDL

• Cache Fill Traffic
  • Temporary content renewal
  • Triggered by revalidation
  • First Fill Request

• Eviction
  • Popularity Profile
  • Least Recently Used

• Topology Considerations
  • Hierarchical Caching gains
  • Geographical Content Affinity

Different Nodal Cache Hit Rates !!!
**Nodal Cache Hit Rates**

- **Probability of a Cache Hit**
  
  $P_{CHR}$

- **Probability of a Nodal Cache Miss**
  
  $P_{CM} = 1 - P_{CHR}$

- **Node A Cache Miss**
  
  $P_{CHR_a} = \frac{3}{5} = 0.60$ or 60%
  
  $P_{CM_a} = 1 - 0.60 = 0.4$

- **Node B Cache Miss**
  
  $P_{CHR_b} = \frac{8}{13} = 0.616$ or 61.6%
  
  $P_{CM_b} = 1 - 0.616 = 0.384$

---

**Content Requests**

- Origin
  
  $\frac{2}{2} = 100\%$

- Cache
  
  $\frac{3}{5} = 60\%$

- Cache
  
  $\frac{8}{13} = 61.6\%$

---

**New Content / Unpopular Content**

**Some Popular Content**

**Very Popular Content**

---

<table>
<thead>
<tr>
<th>Content Requests</th>
<th>Origin</th>
<th>Cache (a)</th>
<th>Cache (b)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>4</td>
<td>5</td>
<td>13</td>
</tr>
</tbody>
</table>
System Cache Hit Rates

• Probability of System Cache Miss
  \[ P_{SCM} = P_{CMa} \times P_{CMb} \]

• System Cache Miss Example
  \[ P_{CHRa} = \frac{3}{5} = 0.60 \text{ or } 60\% \]
  \[ P_{CMa} = 1 - 0.60 = 0.4 \]
  \[ P_{CHRb} = \frac{8}{13} = 0.616 \text{ or } 61.6\% \]
  \[ P_{CMb} = 1 - 0.616 = 0.384 \]
  \[ P_{SCM} = 0.4 \times 0.384 = 0.154 = 15.4\% \]

• System Cache Hit Rate = 1 - 15.4% = 84.6%
Caching Cost: Bandwidth

Cost

Demand

Contributions / Cache-fill

Bandwidth Costs

Source
Data Center
Network Core
Network Edge
Access Network
Home Network
Caching Cost: Cache Storage

Source  Data Center  Network Core  Network Edge  Access Network  Home Network

Cost

Storage
Caching Cost Inflection Point: Optimized Costs

Storage + Bandwidth = Total Cost

Cost

Contributions / Cache-fill

Storage Costs

Bandwidth Costs

Demand

Source Data Center Network Core Network Edge Access Network Home Network

Optimal Costs !!!

Optimal Performance ???


Demand
Caching Architecture
Distribute ‘Enough’ Cache Storage at the ‘Right’ Locations
Content Delivery Functions
Cisco CDN Architecture Evolution

- Encoders
- OMD Cache-Nodes
- vDCM
- Infinite Video
- cDVR MOS
- Cloud Native
- Applications
- Appliance Software
- Service Chains
- Virtualize Software
- Workflow Orchestration
- Elasticity

Time:
- 2011
- 2012
- 2013
- 2014
- 2015
- 2016
- 2017
- 2018

© 2018 Cisco and/or its affiliates. All rights reserved. Cisco Public
Elastic Platform

Video Services Workflow Orchestration

OpenStack – Compute, Storage, and Network Orchestration

Hypervisor

Unified Compute System

Traffic Control

Hypervisor

Unified Compute System

Traffic Caching

Hypervisor

Unified Compute System

Traffic Monitoring

Network Infrastructure
### CDN Functional Architecture

#### Server

<table>
<thead>
<tr>
<th>Server</th>
<th>Function Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traffic Server</td>
<td>HTTP/S proxy-cache based on open source</td>
</tr>
<tr>
<td>Traffic Router</td>
<td>Cache Selection, redirects clients to best cache</td>
</tr>
<tr>
<td>Traffic Ops</td>
<td>Configure and Monitor Traffic Server Caches</td>
</tr>
<tr>
<td>Traffic Vault</td>
<td>Protects keys and certificates</td>
</tr>
<tr>
<td>Traffic Monitor</td>
<td>Monitors cache health and load</td>
</tr>
<tr>
<td>Traffic Stats</td>
<td>Aggregates and visualizes real-time and historical</td>
</tr>
<tr>
<td></td>
<td>performance data</td>
</tr>
<tr>
<td>Influx DB</td>
<td>Time series database to retain all Traffic Server</td>
</tr>
<tr>
<td></td>
<td>statistics</td>
</tr>
<tr>
<td>OMD Analytics</td>
<td>Splunk based CDN Insights with 300+ Dashboards</td>
</tr>
</tbody>
</table>
Traffic Router

- Delivery Service Properties
- Localization using Coverage Zone Map (CZM)
- DNS Content Routing
  - Authoritative DNS for Origin
  - List of Edge Cache IP
- HTTP Content Routing
  - DNS Resolves to Self
  - HTTP Redirect
Traffic Server

• Reverse Proxies
  • URL Re-mapping Logic to Specific Sources (Mid-Tier Cache or Origin)
  • Cache Groups
  • Mid-Tier Cluster (single geographical coordinate)
  • Edge Cluster (single geographical coordinate)

• HTTP Content Routing
  • DNS Resolves to Self
  • HTTP Proxy
Delivery Service

- Cache in RAM, cache on disk, or do not cache at all
- Use DNS or HTTP Content routing
- Limits on transactions per second and bandwidth
- Protocol (http or https)
- Token based authentication settings
- Header rewrite rules
Traffic Monitor

- Throughput (e.g. bytes in, bytes out, etc).
- Transactions (e.g. number of 2xx, 3xx, 4xx responses, etc).
- Connections (e.g. from clients, to parents, origins, etc).
- Cache performance (e.g.: hits, misses, refreshes, etc).
- Storage performance (e.g.: writes, reads, frags, directories, etc).
- System performance (e.g: load average, network throughput, etc).
Traffic Stats

- Cache Stats
  - Bandwidth, Max Kbps, Client Connections

- Delivery Service Stats
  - Service rates Kbps, Status (4xx/5xx), TPS (2xx, 3xx, 4xx, 5xx, total)
  - By Cache-Group, CDN, Delivery Service

- Daily Stats
  - Max Bandwidth, Bytes Served, by CDN
Traffic Vault

- SSL Certificates
  - Private Key, Certificate (CRT), Certificate Signing Request (CSR)

- DNSSec Keys
  - Key Signing Key (private key, public key)
  - Zone Signing Key (private key, public key)

- URL Signing Keys
  - Query Strings
  - Tokens
Content Delivery System Design
Origin Servers and the CDN

- Content Management System (CMS)
  - Combine Content + Publishing Metadata
- Origin Servers (OS)
  - Organized Media on Storage
  - Authorize Mid-Tier Cache
  - Package Content
Origin Server

- Ingest must be flexible, resilient and secure
- CDN can ingest from multiple Origin Servers
  - Local or Remote locations
- Origins can be replicated
  - Locally (load balancing)
  - Remotely (disaster recovery)
- Origins can have structure
  - Security
  - Capture/Recording/Playout separation for better scalability
Delivery Service Content Routing

- Request Redirection model
  - Traffic Router is the Authoritative DNS for “Delivery Service” FQDN
- HTTP-based 30x redirection
  - Traffic Router resolves domain name to its own IP address
  - Traffic Router then uses 302/307 redirection to an Edge Cache
- DNS-based redirection
  - Traffic Router resolves domain to IP address of Edge Cache
- Traffic Router Criteria
  - Based on Client IP Address
Traffic Server Assignment

• Assessing Location (Latency)
  • Per Delivery Service
  • Per Location

• Assessing Status (Availability)
  • Analytics from Edge Caches
  • Resources Available

• Assessing Content Affinity (Performance)
  • Assign Request to Previously Assigned Edge Cache

• Assessing Content Controls
  • Quotas
  • Thresholds
Static Location-based Routing

- Separate Content Routing Plane
  - Implemented at Traffic Router
  - Reference Location Information (MaxMind)

- Traffic Server’s inform Traffic Monitor about status and load using keep-alive messages
  - Server Redundancy

- Variety of Traffic Server Selection criteria available
  - Load
  - Content
  - Service availability
Content Delivery

• Content Affinity Traffic Routing
  • Hash Calculated on URL (HTTP Only)
  • Common URL requests have affinity to same Traffic Server

• Traffic Server Selection
  • Hash Calculated on Origin URL
  • Common Cache-fill requests have affinity to same Traffic Server

• Origin Selection
  • Same as above
Content Delivery Optimization

- Origin Server Sizing depends on CDN Cache Hit Rate (CHR) efficiency
- Define CDN topology and apply Hierarchical Caching to achieve efficiency goal
- Example
  - CDN Efficiency goal: 90%
  - Two-tier CDN (edge + mid-tier-cache)
  - Edge CHR (eCHR): 80%
  - Mid-tier Cache CHR (mCHR): 50%
  - Efficiency =
    - $1 - (1 - eCHR)*(1 - mCHR) = System CHR$
    - $1 - (1 - 0.80)*(1 - 0.50) = 90\%$
Edge Throughput: Peak Busy Hour (PBH)

### Live Load

**Criteria**
- 50 live channels; all profiles
- Bitrate per Channel = 10 Mbps
- Site Demand 30 Gbps

**Edge Live Cache-Fill**
- 50 live * 10 Mbps/live = 0.5 Gbps

### VoD Load

**Criteria**
- Site Demand 20 Gbps
- Edge PoP CHR = 80%

**Edge VoD Cache-fill**
- Load * (1 - CHR)
- 20 Gbps * (0.2) = 4 Gbps

### VoD Load

**Criteria**
- Site Demand 10 Gbps
- Edge PoP CHR = 80%

**Edge VoD Cache-fill**
- Load * (1 - CHR)
- 10 Gbps * (0.2) = 2 Gbps

---

**Site Demand 30 Gbps**

**Edge Live Cache-Fill**
- 50 live * 10 Mbps/live = 0.5 Gbps

**Criteria**
- Site Demand 15 Gbps
- Edge PoP CHR = 80%

**Edge VoD Cache-fill**
- Load * (1 - CHR)
- 10 Gbps * (0.2) = 2 Gbps
Origination Throughput: Peak Busy Hour (PBH)

**Live**

**Criteria**
- 50 live channels
- Bitrate per Channel = 10 Mbps

**Edge Live Cache-Fill**
- 50 live * 10 Mbps/live = 0.5 Gbps

**VoD**

**Criteria**
- Cache Demand 2+4 = 6 Gbps
- Cache CHR = 50%

**Edge VoD Cache-fill**
- Load * (1 - CHR)
- 6 Gbps * (0.5) = 3 Gbps
CDN Characterization: Adaptive Bit Rate
Common Adaptive Bit Rate (ABR) Methods

- **Apple**
  - HTTP Live Streaming (HLS)
  - Segmented H.264 (MPEG2TS)
  - Manifest (.m3u8)

- **Microsoft**
  - HTTP Smooth Streaming (HSS)
  - Fragmented H.264 (MP4)
  - Server Manifest, Client Manifest (.ims, .imsc)

- **MPEG-DASH**
  - Dynamic Adaptive Streaming over HTTP (DASH)
  - Fragmented H.264 (ISO-BMFF)
  - Media Presentation Description (.mpd)
The Challenges with Distributing ABR Objects

- Short fragment / segment sizes ➔ High HTTP Request Rate
- URL’s can be Absolute or Relative ➔ DNS Resolutions
- TCP connections should not be short-lived (client code) ➔ Pipeline HTTP Requests
- CDS object handling configured on a per Delivery Service basis

Progressive Download

<table>
<thead>
<tr>
<th>Time</th>
<th>Start GET</th>
<th>+ 2 sec GET</th>
<th>+ 4 sec GET</th>
<th>+ 6 sec GET</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frag1-1</td>
<td>512 kbps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frag2-1</td>
<td>768 kbps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frag3-1</td>
<td>1.0 mbps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frag4-1</td>
<td>1.5 mbps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frag1-2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frag2-2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frag3-2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frag4-2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frag1-3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frag2-3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frag3-3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frag4-3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frag1-4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frag2-4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frag3-4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frag4-4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Movie.mp4

- 2hr movie, 2 sec segments
- 3600 fragments x 7 profiles
- 25,000 objects/movie
Live / Linear TV: HLS

- Caching the Sliding Window
  - Time-to-Live (TTL) access to Manifest / Profiles (10s)
  - Time-to-Live access to Segments (30s)
Live / Linear TV: DASH and HSS

- Caching the Time-shift Window
  - Time-to-Live (TTL) access to .imsc Manifest (2s)
  - Repackaged .mpd (1-10sec)

![Diagram of Live Linear TV DASH and HSS](image.png)

- Current Manifest Available
- Manifest TTL Expired in CDN
- Segment TTL Remaining in CDN
- Segment TTL Expired in CDN
- No Segment in CDN

SOURCE

MPEG2TS

Transcode / Package

TTL remaining

TTL Expired

T -4  T -2  TNOW  T +2  T +4  T +6  T +8  T +10

HSS Profile 5
HSS Profile 4
HSS Profile 3
HSS Profile 2
HSS Profile 1

Client 1
Client 2
VoD: HLS

- Complete Mapping of Manifest to Segments
- Time-to-Live of Manifest / Profiles (as specified)
- Time-to-Live access to Segments (as specified)

---

**Diagram:**

- **segment.ts** → **manifest.m3u8**
- **Transcode / Package**
- **MPEG2TS**

---

**Timeline:**

- **T_{NOW}**
- **T_{+10}**
- **T_{+20}**
- **T_{+30}**
- **T_{+40}**
- **T_{+50}**

---

**Legend:**

- **Manifest TTL Remaining in CDN**
- **Manifest TTL Expired in CDN**
- **Segment TTL Remaining in CDN**
- **Segment TTL Expired in CDN**
- **No Segment in CDN**

---

**Note:**

- Complete Manifest
- Cached until TTL Expired

---

**Client:**

- **Client 1**
- **Client 2**

---

**Profiles:**

- **HLS Profile 5**
- **HLS Profile 4**
- **HLS Profile 3**
- **HLS Profile 2**
- **HLS Profile 1**
VoD: HSS and DASH

- Caching the ‘DVR Duration’ Window
  - Time-to-Live (TTL) access to .imsc Manifest (2s)
  - Time-to-Live access to Fragments (6s – IIS default 2hr)
## The Challenges with Distributing ABR Objects

### Transaction Rates: Transactions Per Second for 1HR asset

<table>
<thead>
<tr>
<th>Behavior</th>
<th>Object Length (sec)</th>
<th>Client Request (TPS)</th>
<th>2000 clients (TPS)</th>
<th>Asset Requests (Objects/Hour)</th>
<th>200 channels (Objects / Hour)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smooth</td>
<td>2</td>
<td>0.5000</td>
<td>1,000</td>
<td>1800</td>
<td>360,000</td>
</tr>
<tr>
<td>HLS</td>
<td>10</td>
<td>0.1000</td>
<td>200</td>
<td>360</td>
<td>72,000</td>
</tr>
<tr>
<td>PDL</td>
<td>3600</td>
<td>0.0003</td>
<td>0.56</td>
<td>1</td>
<td>200</td>
</tr>
</tbody>
</table>

### Cache Object Size (MB)

<table>
<thead>
<tr>
<th>Bit Rate</th>
<th>3000 kbps</th>
<th>1500 kbps</th>
<th>500 kbps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smooth</td>
<td>0.75</td>
<td>0.38</td>
<td>0.13</td>
</tr>
<tr>
<td>HLS</td>
<td>3.8</td>
<td>1.9</td>
<td>0.6</td>
</tr>
<tr>
<td>PDL</td>
<td>1,350</td>
<td>675</td>
<td>225</td>
</tr>
</tbody>
</table>

### Origin File Count (One Hour Asset)

<table>
<thead>
<tr>
<th>Bit Rate</th>
<th>3000 kbps</th>
<th>1500 kbps</th>
<th>500 kbps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smooth</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>HLS</td>
<td>360</td>
<td>360</td>
<td>360</td>
</tr>
<tr>
<td>PDL</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

- **High Transaction Rate**: 
  - Smooth: 360,000
  - HLS: 72,000
  - PDL: 200

- **Small Object Write/Read Performance**: 
  - Smooth: 0.13
  - HLS: 0.6
  - PDL: 225

- **File Object Count**: 
  - HLS: 360
# HSS vs. HLS: Vastly Different Methods

<table>
<thead>
<tr>
<th></th>
<th>HSS (Microsoft)</th>
<th>HLS (Apple)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transport Protocol</td>
<td>HTTP</td>
<td>HTTP</td>
</tr>
<tr>
<td>Fragment Size (default)</td>
<td>2 seconds</td>
<td>10 seconds</td>
</tr>
<tr>
<td>#TCP connections</td>
<td>2</td>
<td>1-2</td>
</tr>
<tr>
<td># Content Files on Origin Server</td>
<td>#profiles</td>
<td>#profiles x 360/Hr. of content</td>
</tr>
<tr>
<td>Codec Support</td>
<td>VC-1, H.264, WMA</td>
<td>H.264</td>
</tr>
<tr>
<td>Wire Format</td>
<td>MP4 fragments</td>
<td>MPEG2TS video segments, audio ID3, webVTT</td>
</tr>
<tr>
<td>File Format on Origin Server</td>
<td>.ismv, Fragmented mp4</td>
<td>.ts, Segmented TS</td>
</tr>
<tr>
<td>Standard HTTP Origin Server</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Encryption/DRM</td>
<td>PlayReady; AES-128 in Counter Mode</td>
<td>AES-128 in Cipher Block Chaining</td>
</tr>
<tr>
<td>Client</td>
<td>Silverlight, OSMF (OpenSource)</td>
<td>iPhone OS +, QuickTime X</td>
</tr>
<tr>
<td>Client Manifest file</td>
<td>.ismc (.ism/manifest or .isml/manifest)</td>
<td>.m3u8</td>
</tr>
<tr>
<td>Origin server</td>
<td>Helper integrated with IIS server</td>
<td>HTTP server</td>
</tr>
</tbody>
</table>
# HSS vs. DASH: DASH is a super-set of HSS

<table>
<thead>
<tr>
<th></th>
<th>HSS (Microsoft)</th>
<th>DASH (MPEG DASH Guidelines v3)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Transport Protocol</strong></td>
<td>HTTP</td>
<td>HTTP</td>
</tr>
<tr>
<td><strong>Fragment Size (typical)</strong></td>
<td>2 seconds</td>
<td>1-10 seconds</td>
</tr>
<tr>
<td><strong>#TCP connections</strong></td>
<td>2</td>
<td>1, 2</td>
</tr>
<tr>
<td><strong># Content Files on Origin Server</strong></td>
<td>#profiles</td>
<td>#programs</td>
</tr>
<tr>
<td><strong>Codec Support</strong></td>
<td>VC-1, H.264, WMA</td>
<td>H.264, H.265</td>
</tr>
<tr>
<td><strong>Wire Format</strong></td>
<td>MP4 fragments</td>
<td>MP4 fragments</td>
</tr>
<tr>
<td><strong>File Format on Origin Server</strong></td>
<td>.ismv, Fragmented mp4</td>
<td>ISO-BMFF</td>
</tr>
<tr>
<td><strong>Standard HTTP Origin Server</strong></td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Encryption/DRM</strong></td>
<td>PlayReady; AES-128 in Counter Mode</td>
<td>Common Encryption, AES-128, CTR+ CBC</td>
</tr>
<tr>
<td><strong>Client</strong></td>
<td>Silverlight, OSMF (OpenSource)</td>
<td>DASH Client</td>
</tr>
<tr>
<td><strong>Client Manifest file</strong></td>
<td>.ismc (.ism/manifest or .isml/manifest)</td>
<td>.mpd (media presentation description)</td>
</tr>
<tr>
<td><strong>Origin server</strong></td>
<td>Helper integrated with IIS server</td>
<td>Packager (MPD), Server (Segments)</td>
</tr>
</tbody>
</table>
CDN Optimizations for ABR

- Optimized TCP connection handling
  - Scaling to support the large # of connections for ABR

- Optimized HTTP request handling
  - Scaling to support the large # of GET requests for ABR

- Request Bundling
  - For live streaming, aggregates multiple cache-fill requests for the same content into a single request

- Small Object Cache Throughput Optimizations
  - Small objects written to memory, delayed write to disk (SSD or HDD)
  - Large objects continue to be cached on disk
  - Client/Cache Stickiness (Content Affinity)

- Service Visibility
  - Reporting and Analytics optimizations for ABR
New Approaches Caching
Anatomy of a URL

IPv4 ➤ IPv6
• Scale by 30 orders of magnitude
• IPv6 object representations
Encoding of a Media Object

![Table](image)

IPv6: Service Prefix | IPv6: Media Object
New Approaches

**6CN – IPv6 Content Networking**
- Mapping URL’s into IPv6 Addresses
- Classic CDN Route Redirection
- Retaining classic TCP/IP transport

**ICN – Information Centric Networking**
- Named-data Objects (segment names without addresses)
- New named-routing infrastructure
- Elimination of Transport Layer

**hICN – Hybrid Information Centric Networking**
- Mapping URL’s into IPv6 Addresses
- IPv6 routing of “named-data”
- Transport Layer = Metadata
## Naming and Transport Methods

<table>
<thead>
<tr>
<th>6CN</th>
<th>Hybrid ICN</th>
<th>ICN</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Names into IPv6 addresses</strong></td>
<td><strong>Names into IPv6 addresses</strong></td>
<td><strong>Variable length routable names</strong></td>
</tr>
<tr>
<td><strong>L4-7 request routing based on names (e.g. with SR)</strong></td>
<td><strong>L3 Name-based routing and h2h dynamic forwarding</strong></td>
<td><strong>L3 Name-based routing and h2h dynamic forwarding</strong></td>
</tr>
<tr>
<td><strong>Connection-based sender-driven transport</strong></td>
<td><strong>Partially symmetric routing</strong></td>
<td><strong>Symmetric routing</strong></td>
</tr>
<tr>
<td><strong>Tunnel-based security</strong></td>
<td><strong>Connectionless receiver-driven multipath transport</strong></td>
<td><strong>Connectionless receiver-driven multipath transport</strong></td>
</tr>
<tr>
<td><strong>Anchor-based mobility</strong></td>
<td><strong>Object-based security</strong></td>
<td><strong>Object-based security</strong></td>
</tr>
<tr>
<td><strong>Application-layer (CDN) proactive caching</strong></td>
<td><strong>Anchorless mobility</strong></td>
<td><strong>Anchorless mobility</strong></td>
</tr>
<tr>
<td></td>
<td><strong>In-path reactive caching</strong></td>
<td><strong>In-path reactive caching</strong></td>
</tr>
</tbody>
</table>
6CN Routing directly to IPv6 Content

- Identify each video chunk with a unique IPv6 address.
- Instead of: http://cdn.example.com/video?cid=1af429fac&profile=12&[...]

<Representation bandwidth="3146352" codecs="avc3.640032" frameRate="31" SegmentList duration="58050" timescale="30000">
  <SegmentURL media="http://[2604:1380:1000:86ed:4000:100:691:3]" />
</Representation>
Information Centric Networking
A new paradigm in communications

• Definitions:

  • NAMED-DATA
    • uniquely identified chunk of data, state, or results of a process
    • {e.g. sensor reading, calculation, or .. a media fragment}

  • PRODUCER
    • Produces named-data

  • CONSUMER
    • Creates “Interest” requests for named-data

  • FORWARDER
    • Forwards named-data object along ”Interest” path
Hybrid Information Centric Network

Network-Native Video Distribution

1. Named-Data Content Packaging
2. Named-Data Objects Ingested
3. In-Network Caching from SP to Subscriber
4. Net-Native Direct (not redirect) IP Content routing
5. hICN Network Analytics become Content Aware

Remote Production
Contribution
Studio & Post Production
Content Preparation & Playout
Content Push + Linear/Live “Channels”
Video Streaming Content Pull
Consumer Experience

Acquire/Create
hICN Packaging
hICN Distribution
Platform Optimization and Orchestration
Platform Requirements

• Three Dimensions of Analysis
  • Compute
    • Computation Complexity
    • Transactions Per Second
  • Storage
    • Volume of Data
    • Read / Write Capabilities of Storage
  • Bandwidth
    • Throughput
    • Packets Per Second
System Load Requirements: Linear Unicast

Assumptions:
- Stream Demand: 1500 Gbps
- Channels: 500
- Size: 10Mbps per channel
- Caching: In-memory
System Load Requirements: Video On Demand

Assumptions:
- Stream Demand: 500 Gbps
- Titles: 200,000
- Size: 3.2 GB per Title (avg 43min)
- Caching: 97% CHR
System Load Requirements: Cloud DVR

Assumptions:
- Stream Demand: 1320Gbps
- Titles: Unique Copy
- Size: 3.2 GB per Title (avg 43 min)
- Caching: 0% CHR
System Load Requirements: Combined Services

Data Center Services

Compute (CPU)
Storage (TB)
Bandwidth (Gbps)

Streaming
Origination
Manipulation
Encryption
Encoding

Bandwidth (Gbps)
Storage (TB)
Compute (CPU)
Example: Cloud-Enabled Media Origination Services
MOS Controller – High Level Modules

Control Plane

Management functionality
Service Manager, GUI Manager, Document Services

Media Control Plane
Application Instance Controller (AIC)
Content Object Store Controller (COS AIC)
Centralized Logging (ELK)

Service Orchestration
Service Instance Controller (SIC)
Asset Workflow Manager (AWM)

Data Plane

Media Data Plane
MCE, MPE (MPE-C, MPE-TC)
App Engines (State Cache HA-Proxy, IPVS)
Unified Media Components Communication

Ingest
- OpenStack SWIFT Storage
  - MCE
  - MCE
  - MCE

Redistribution Key Lookup
- Key Mgmt
- HA Proxy
- Key Mgmt
- HA Proxy

Playback Request (DASH)
- LB
- MPE
- MPE
- MPE

Playback Request (HLS, HSS, HDS)
- Playback Request
- Redirect to MPE

Playback Response (DASH)
- Playback Response
- Playback Request (DASH)
- Playback Request (DASH)
- Playback Request (HLS, HSS, HDS)

Cisco
Linux
3rd Party
Key Points
Key Takeaways: Caching

• CDN Value Proposition
  • Efficient Distribution of Content (Audio, Media, Software)
  • Improved Performance / Scale
  • Resiliency

• Content Caching Methods
  • Emphasis on Proxy Caching (Authorized) vs Transparent
  • Leveraging Two Referral Methods: DNS-based or HTTP Redirect

• CDN Architectural Choices
  • Strategic Hierarchical Caching
  • Cost Optimization: Bandwidth Versus Storage
Q & A
Cisco Spark

Questions?
Use Cisco Spark to communicate with the speaker after the session

How
1. Find this session in the Cisco Live Mobile App
2. Click “Join the Discussion”
3. Install Spark or go directly to the space
4. Enter messages/questions in the space

cs.co/ciscolivebot#BRKSPV-2160
• Please complete your Online Session Evaluations after each session

• Complete 4 Session Evaluations & the Overall Conference Evaluation (available from Thursday) to receive your Cisco Live T-shirt

• All surveys can be completed via the Cisco Live Mobile App or the Communication Stations

Don’t forget: Cisco Live sessions will be available for viewing on-demand after the event at www.ciscolive.com/global/on-demand-library/.
Continue Your Education

• Demos in the Cisco campus
• Walk-in Self-Paced Labs
• Tech Circle
• Meet the Engineer 1:1 meetings
• Related sessions
Thank you