Understanding Video Management Planes

Zahaib Akhtar*
University of Southern California

Yun Seong Nam*
Purdue University

Jessica Chen
University of Windsor

Ramesh Govindan
University of Southern California

Ethan Katz-Bassett
Columbia University

Sanjay Rao
Purdue University

Jibin Zhan
Conviva

Hui Zhang
Conviva

ABSTRACT

While Internet video control and data planes have received much research attention, little is known about the video management plane. In this paper, using data from more than a hundred video publishers spanning two years, we characterize the video management plane and its evolution. The management plane shows significant diversity with respect to video packaging, playback device support, and CDN use, and current trends suggest increasing diversity in some of these dimensions. This diversity adds complexity to management, and we show that the complexity of many management tasks is sub-linearly correlated with the number of hours a publisher's content is viewed. Moreover, today each publisher runs an independent management plane, and this practice can lead to sub-optimal outcomes for syndicated content, such as redundancies in CDN storage and loss of control for content owners over delivery quality.

CCS CONCEPTS

• **Information systems** → **Multimedia streaming**; *Multimedia content creation*; • **Networks** → **Network measurement**;

KEYWORDS

Video Delivery, Video Management Plane

1 INTRODUCTION

Video forms the overwhelming majority of Internet traffic [29, 36, 39, 42]. The deluge in video traffic is due both to the popularity of large services like YouTube, Netflix, and Facebook [20, 23, 35] and to the

significant increase in Internet video services provided by publishers who traditionally produced content for broadcast television [43].

An Internet video publisher must (i) split the video into chunks, encode each chunk at one or more bitrates, and encapsulate chunks using a *streaming protocol*; (ii) develop and maintain playback software for the wide range of user devices; and (iii) distribute video to Content Delivery Networks (CDNs). We refer to these tasks as *video management plane* operations (§2), as distinct from *control plane* operations that involve selecting which CDN to direct a user to and what bitrate to choose for each chunk, and *data plane* operations that involves transporting each chunk to the end user. Whereas the data and control planes have received much attention (e.g., [47, 61, 65, 66, 71, 73, 76, 80]), video management plane decisions have been relatively unexplored, even though they impact how many users and devices a publisher can reach, the computation and storage requirements of content publishers, the complexity of troubleshooting, application performance, and the effort needed to incorporate control plane innovations such as new bitrate selection algorithms [65, 68, 71, 76, 80].

Our paper characterizes aspects of video management planes for more than 100 content publishers (§3), including 7 of the top 10 subscription video publishers [11], as well as prominent sports and news broadcasters and on-demand video publishers. Our dataset comes from Conviva [9], which is a streaming TV measurement and intelligence platform. The dataset contains metadata for over 100 billion video views, including metadata about the client (e.g., device and application used), video (e.g., anonymous publisher ID and video ID) and delivery (CDN, performance metrics). The aggregate daily view-hours across all our publishers are comparable to reported values for Facebook and Netflix.

Two aspects make our data unique relative to published industry reports [11, 13, 41] (§7). Our data spans 27 months, enabling analysis of management plane practices over time. It also lets us assess *view-hours* (the total number of hours content is viewed) and *views* (the total number of video sessions) for any slice of the data (*e.g.,* how many view-hours or views can be attributed to mobile apps).

Contributions. First, we characterize video management planes along three key dimensions (§4): streaming protocols, playback devices and platforms, and CDNs. For each dimension, we characterize (i) how each instance (*e.g.,* a specific streaming protocol, or a specific platform category such as the set-top box) has evolved across publishers, and over time; and (ii) the number of instances of each dimension used by a given publisher and its evolution, and how this correlates with the publisher's view-hours.

---
* Both authors contributed equally to this paper and can be contacted at following: zakhtar@usc.edu, nam21@purdue.edu

Several common themes run across our analysis of these three dimensions. We find that, despite significant changes over the 27 month period, *no single dominant alternative* has emerged along any dimension. Among streaming protocols, HLS and DASH have significant usage, while view-hours are almost evenly distributed across 3 CDNs and across 2 platforms (browser and mobile). Moreover, more than 90% of view-hours can be attributed to publishers who support more than 1 protocol. The same is true of publishers who use more than 1 CDN, and publishers who support more than 1 platform. Publishers with more view-hours tend to support more choices of protocols, platforms, and CDNs.

Beyond these, our analysis uncovers some new findings: streaming set-top boxes[1] dominate by view-hours; almost 80% of view-hours are from publishers that support 4-5 CDNs; and a significant fraction of publishers who use multiple CDNs segregate live and on-demand traffic by CDN. Our analysis also adds color to known findings. For example, DASH usage has increased, but this growth is being driven entirely by large publishers. Moreover, while mobile app views have indeed increased, view-hours have not proportionally increased because view durations on mobile devices tend to be short.

Second, we take an initial step towards quantifying the impact of three dimensions of diversity on the *complexity* of management plane operations such as software maintenance, failure triaging, and packaging overheads (§5). We find that metrics that approximate the complexity of these operations for a publisher are sub-linearly correlated with the publisher's view-hours. For example, a publisher with $10\times$ as many view-hours as another will tend to maintain $1.8\times$ as many versions of its video playback software.

Third, we demonstrate that today's management plane practices may not be well suited for content *syndication* (§6), in which *syndicators* license and redistribute content from a *content owner*. Syndication is prevalent in Internet video, yet syndicators run video management planes that are independent from those of content owners. As a result, we find cases where, for the *same* content, owners' clients observe significantly different delivery performance than syndicators' clients. We also find that more integrated management planes between owners and syndicators can reduce CDN origin storage requirements for a popular video series by $2\times$.

Our results further our understanding of video management planes and open the door for research into new syndication models, complexity metrics, and approaches to cope with diversity and reduce management complexity.

2 THE VIDEO MANAGEMENT PLANE

A video *publisher* makes available online live and/or stored video content. Video content is encoded in different *formats*, delivered by one or more *CDNs*, and delivered to *playback software* on user devices. The video control and data planes together achieve *chunked adaptive streaming*: the data plane streams video *chunks* over HTTP, and the control plane adaptively determines, based on network conditions, which *bitrate* a chunk is downloaded at, and from which CDN.
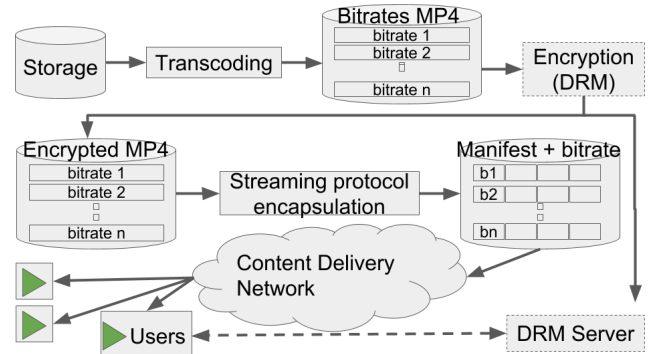

Figure 1: A video delivery pipeline.

Each publisher operates a *video management plane*, a term we use for a pipeline of automated systems (some with humans in the loop), Fig. 1 shows one such pipeline, that perform two primary functions. The first function *prepares* video content for delivery to users. Preparation involves *packaging* the video content and *distributing content to CDNs* for delivery to users. The second function is to *develop and maintain* playback software for the wide range of devices on which video is consumed by users.[2]

Packaging. Packaging achieves two goals: 1) preparing content for adaptive streaming and 2) generating the necessary information for an end user device to perform playback.

Encoding. The first packaging step transcodes the master video file into multiple bitrates of encodings such as H.264 [15], H.265 [16] or VP9 [32]. A video bitrate encodes the video at a certain *resolution* and a certain *quality*. A given resolution can be encoded at different qualities, which differ in the degree of lossy compression applied to trade-off perceptual quality for reduced bandwidth. Publishers optionally use *DRM* (Digital Rights Management) software to encrypt the video so that only authenticated users can access it.[3]

Each encoded bitrate of the video is then broken into chunks (a chunk is a fixed playback-duration portion of the video) for adaptive streaming and encapsulated using a *Streaming Protocol* (discussed below). Some publishers support byte-range addressing, where clients can request an arbitrary byte range for a given bitrate instead of chunks.

Streaming protocols. Streaming protocols define the encapsulation format for video chunks to enable delivery over the network. A number of streaming protocols are in use today including Apple's HLS [4], Microsoft's SmoothStreaming (MSS [19]), Adobe's HDS [1] as well as an open standard MPEG-DASH protocol (DASH) [12]. Of these, Apple's devices only support HLS, though recent Apple devices allow limited support for DASH [40]. Some protocols like DASH [12] can support any video encoding format, while others like HLS only support a fixed set of codecs [6].

Streaming protocols also specify metadata about the video necessary for adaptation by the control plane. This metadata is stored in a *manifest* file. The manifest contains information about a number

---

[1]Streaming set-top boxes are different from traditional cable set-top boxes. These include devices such as Roku, FireTV, AppleTV etc. Throughout this paper we use the term set-top box to refer to **streaming** set-top boxes.

[2]Other management plane functions, including accounting, billing, and fault isolation, are beyond the scope of the paper.

[3]This is orthogonal to TLS encryption of the video during transmission over HTTPS.

of attributes including the values of available bitrates for adaptation, the audio bitrates, the time duration of an individual chunk and the URLs to fetch video chunks *etc.*

Device Playback. The next function of the management plane is to support the range of devices on which a user can view the publisher's content. To enable playback on them, publishers either provide *Video Players* embedded in web pages to permit browser-based viewing or *Apps* on devices that permit app-based content delivery.

Browser-based video players today are either implemented using JavaScript inserted into webpages using native HTML5 support, or using external plugins such as Flash or Microsoft's Silverlight. However, several types of devices such as streaming set-top boxes (e.g. Roku, AppleTV), game consoles (e.g Xbox), smart TVs (e.g. Samsung TV), and mobile devices use app-based playback. To build these apps, publishers use device-specific SDKs (Software Development Kits, sometimes called Application Frameworks) which provide support for frame rendering, user controls *etc.* as well as bitrate adaptation logic [48, 65, 68, 71, 76, 80]. Because publishers may have to support different devices, and, for a given device, different SDK versions (since users may take time to upgrade their device SDKs), at any given time publishers may have to maintain several versions of their app (one for each device-SDK version combination).

Content Distribution. Publishers employ Content Distribution Networks (CDNs). Some content publishers such as YouTube and Netflix deploy their own CDNs. The publishers in our dataset serve their videos via *third-party CDNs* (though some also use private CDNs). To improve performance and availability, some publishers serve content through multiple CDNs [61, 66, 69]. Some publishers use a CDN broker to select the best CDN for a given client view [72]. Even some publishers who only use a single CDN use a CDN broker for management services such as monitoring and fault isolation.

Most publishers proactively *push* content to CDNs. A publisher may either push packaged chunks to each of its CDNs, or may use a packaging service provided by a CDN. In the latter case, the publisher pushes the master video file (or live video stream), and the CDN performs the packaging on behalf of the publisher. The client playback software retrieves chunks using URLs in the manifest file.

3 GOALS, METHODOLOGY & DATASET

Goals. We want to characterize, at scale, publisher video management plane practices (with respect to packaging, CDN use, and device support) and how they have evolved over time. We also present preliminary analyses to understand the implications of these findings on the complexity of video management, and the performance of video delivery.

Prior industry reports have explored related aspects of video management planes [13, 18, 22, 37, 41] (see §7 for details). These studies have four shortcomings that we address in this paper. First, they lack a *publisher-centric* focus, even though publishers make video management decisions. As a simple example, these studies do not reveal how many streaming protocols or how many CDNs a publisher uses, but these factors affect management complexity (§5). Second, these studies do not *contextualize* their results. For example, consider a finding that few publishers use DASH. If these publishers are large, they are more likely to drive adoption of DASH than if

Protocol	Extension	Sample URL
HLS	.m3u8, .m3u	http://[...].akamaihd.net/master.m3u8
DASH	.mpd	http://[...].llwnd.net//Z53TiGRzq.mpd
SmoothStreaming	.ism, .isml	http://[...].level3.net/56.ism/manifest
HDS	.f4m	http://[...].aws.com/cache/hds.f4m

Table 1: Streaming protocol file extensions and sample URLs

they are all small publishers. Third, most studies were one-off, but the video landscape is continuously evolving. Longitudinal trends can help understand how the video delivery ecosystem is likely to evolve in the short to medium term. Fourth, in part because they lack a publisher-centric focus, these studies do not shed light on a common practice in video delivery, *content syndication* (§6).

Extracting management plane practices from streaming analytics dataset. We use data from Conviva [9], a streaming TV measurement and intelligence platform. Conviva provides various services such as monitoring playback quality and dynamic selection of CDNs. To enable these services, Conviva provides a monitoring library which publishers integrate with their video players. The monitoring library reports per-view information to Conviva's backend which collects data from different devices including desktops, mobiles, and smart TVs.

Our dataset spans 27 months (January 2016 to March 2018) and contains over 100 billion views. For each view, we have access to an anonymized publisher ID, a URL which uses an anonymized video ID but retains the manifest file extension, device model (*e.g.,* iPhone, Roku); the operating system (*e.g.,* iOS, Android); HTTP user-agent (for browser views) or SDK and SDK version number (for app views); the CDN(s) that were used to deliver the content;[4] the set of available bitrates for adaptation; viewing time; and delivery performance (average bitrate and rebuffering time).

From this data, we can extract, for any given time window (*e.g.,* a month), and for each publisher: which CDNs the publisher uses, and which devices the publisher's content was viewed on. We also infer which streaming protocols a publisher uses by examining the manifest file extension. Different streaming protocols use pre-defined file extension types for their manifest files (Tab. 1): for example, HLS manifest files typically use the `.m3u8` file extension[5].

For each of these dimensions, we can associate three measures that can help contextualize the dimension. Our primary measure, and one used most often in the video industry [20, 23, 35], is the number of *view-hours* (*i.e.,* the total viewing time in terms of hours). Using our data, we can examine, for example, the number of view-hours of a publisher's content delivered from a given CDN, over HLS, to iPhones. In some of our analysis, we use *the number of views* associated with a particular dimension (*e.g.,* the number of video plays delivered to Roku players). This is helpful to understand if view-hours were accumulated from a few long video views or many short video views. In some analyses, we also measure importance by the number of distinct videos IDs. We do not have this data for all our publishers, so when we use this measure, it is an under-estimate.

Dataset limitations. Our dataset does not include the 3 largest video publishers: YouTube, Netflix and Facebook. However, where

---

[4]During a single view, chunks may be downloaded from multiple CDNs.

[5]There are two exceptions to this. RTMP can be detected from the protocol specification in the URL (RTMP instead of HTTP). Progressive downloading uses file extensions corresponding to video encodings, like `.mp4` or `.flv`.

possible we use published reports on these large services to contextualize the findings in our study. The dataset also does not include data necessary to investigate three aspects of video management: Digital Rights Management (DRM) usage, monetization, and encoding format.

Macroscopic context. Video consumption today is dominated by YouTube, Facebook, and Netflix, which contributed (according to 2016/2017 studies [21, 30]) 1 billion, 0.1 billion and 0.14 billion view-hours per day respectively. Beyond these three, there are a large number of video publishers that deliver online content. Our March 2018 dataset comprises of more than one hundred publishers. Across all our publishers, the aggregate daily view-hours is 0.06 billion per day, comparable to Facebook and Netflix in 2016 and 2017. Finally, the publishers in our study together serve 180 countries.

4 CHARACTERIZING VIDEO MANAGEMENT PLANES

We characterize video management planes along three dimensions: *packaging*, measured by the streaming protocols used; *content distribution*, measured by the CDNs used; and *device playback*, measured by types of devices and number of application frameworks. For each dimension d, we ask:

- How has d evolved across publishers?
- How has d evolved in terms of view-hours? For example, does a dominant practice result from a few big publishers or many small publishers?
- What is the distribution across publishers of number of instances of d? Is it correlated with publisher view-hours?

Our two-year dataset is too large to process every view, so we use a sequence of two-day snapshots taken bi-weekly. We use the last snapshot, taken in March 2018, for the third question.

4.1 Packaging

Understanding the prevalence of different streaming protocols is important for several reasons. First, the amount of work/resource needed to package content is proportional to the number of streaming protocols supported by a publisher. Also the time taken to package content can add delay to live content distribution. Second, in some cases, support for a streaming protocol can directly impact the set of devices that can be supported: *e.g.,* until recently publishers needed to support HLS to work with Apple devices (§2).

Prevalence by streaming protocol. Streaming protocols include HTTP-based protocols as well as RTMP, a protocol for low latency video streaming services [14, 75, 78]. In our dataset, RTMP only accounted for 1.6% of the view-hours in January 2016 and 0.1% in March 2018. RTMP has compatibility issues with network middleboxes, scalability limitations [75, 78], and limited device support. For these reasons, our publishers prefer HTTP-based streaming protocols even though these protocols may add a few seconds of encoding and packaging delay to live streams. The rest of our analysis focuses on HTTP-based protocols.

Across publishers. Fig. 2(a) shows the percentage of publishers that supported a given streaming protocol over time (the sum of percentages at any given point in time exceeds 100% because publishers can support multiple protocols). The *rightmost point*
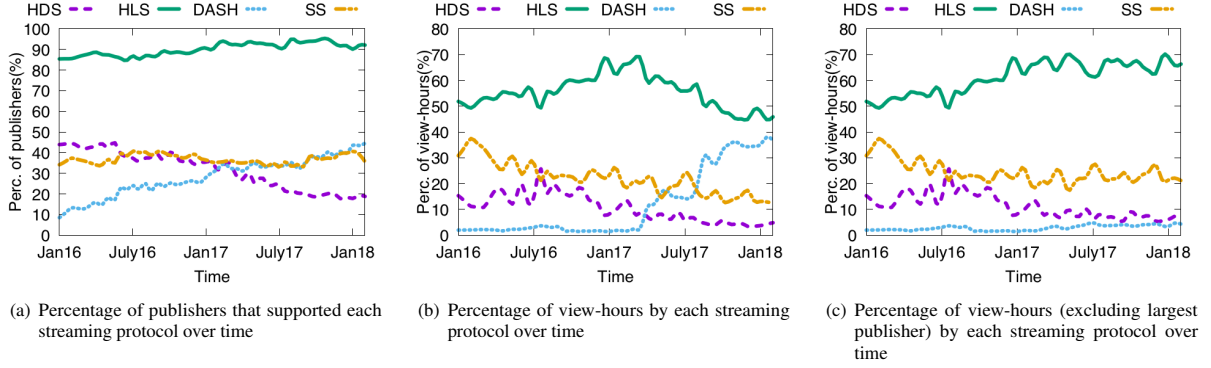
of each curve *indicates the latest snapshot*. In this latest snapshot, 91% of publishers support HLS, likely because many devices and players support it [17, 25, 30]. DASH and SmoothStreaming are currently supported by around 40% of publishers, but HDS is only supported by 19% of the publishers. Over time, support for DASH has increased from 10% of publishers to 43%, corroborating a recent survey of video developers [41]. HDS has steadily lost support. The growth of DASH has not been at the expense of HLS or SmoothStreaming. Over time, HLS and SmoothStreaming support across publishers has remained steady.

By view-hours. We can quantify usage of streaming protocols in terms of *view-hours*, unlike existing industry surveys [13, 41]. Fig. 2(b) shows the percentage of view-hours served by different protocols over time. In our latest snapshot, HLS and DASH are dominant, each accounting for about 38-45% of the view-hours, with the other two being relatively small. Longitudinally, the most noticeable trend is the growth in use for DASH from 3% to 38% view-hours. We found that this is due to N large publishers in our dataset (where N is small and exact number is not provided for privacy concerns). This increase in DASH support noticeably reduces the fraction of view-hours attributable to the other protocols.
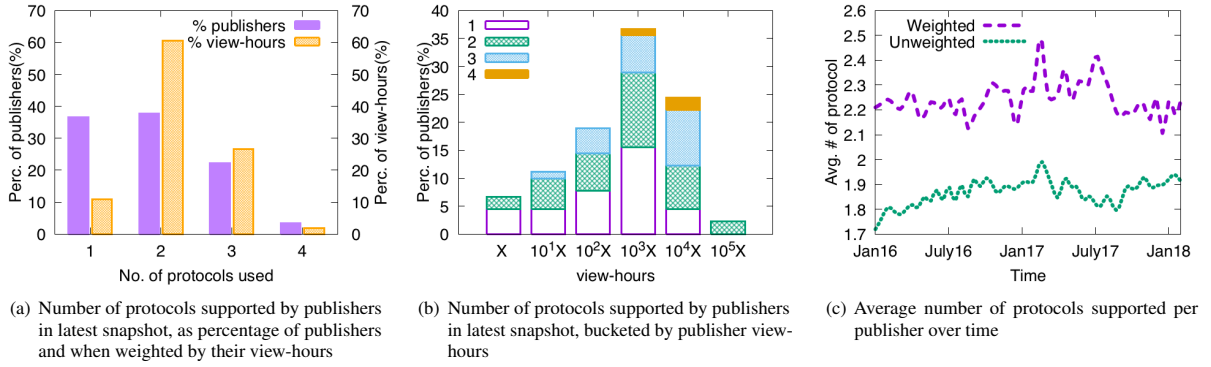
When we remove these publishers (Fig. 2(c)), we observe that DASH support from other publishers only accounts for less than 5% of view-hours overall. To explain this further, Fig. 4 shows the distribution across publishers of the percentage of view-hours that used a given protocol, only considering publishers that support that protocol. Even though 40% of the publishers support DASH (Fig. 2(a)), half of them employ it for at most 20% of their view-hours (Fig. 4). In contrast, among the 90% of publishers that support HLS, half use it for at least 85% of their view-hours.

Number of protocols per publisher. Fig. 3(a) explores *how many* streaming protocols each publisher supports in the latest snapshot. Each group of bars corresponds to a given number of protocols n and shows the percentage of publishers that used n protocols (left) and the percentage of view-hours from publishers that used n protocols (right). While 38% of publishers support 1 protocol, these publishers account for less than 10% of view-hours. The use of 2 protocols is dominant (38% of publishers, accounting for nearly 60% of view-hours), and the use of 3 protocols is also significant.

Fig. 3(b) presents the number of protocols used by publishers when bucketed by their view-hours. The left most bar corresponds to publishers with X view-hours or less (we do not specify X for confidentiality reasons). The second bar corresponds to publishers with X to $10X$ daily view-hours, the next bar to publishers with $10X$ to $100X$ view-hours, and so on. Each bar corresponds to the percentage of publishers in a given bucket, broken down by the number of protocols used by the publishers. The tallest bar indicates that (i) over 35% of publishers have $100X$ to $1000X$ daily view-hours; and (ii) publishers in this bucket use from 1 to 4 protocols. Outside of publishers in the bucket with the least view-hours, more than 50% of publishers in all buckets use at least 2 protocols, with all publishers in the $10^4 X - 10^5 X$ bucket (right-most bar) using 2 protocols. A significant number of publishers in the intermediate buckets use 3 or 4 protocols.

(a) Percentage of publishers that supported each streaming protocol over time

(b) Percentage of view-hours by each streaming protocol over time

(c) Percentage of view-hours (excluding largest publisher) by each streaming protocol over time

Figure 2: Streaming protocols used in terms of percentages of publishers and view-hours for past 27 months

(a) Number of protocols supported by publishers in latest snapshot, as percentage of publishers and when weighted by their view-hours

(b) Number of protocols supported by publishers in latest snapshot, bucketed by publisher view-hours

(c) Average number of protocols supported per publisher over time

Figure 3: Number of streaming protocols used by publishers (by % of publishers and by their view-hours).

Fig. 3(c) shows changes in the number of protocols used over time. The lower curve shows the number of protocols averaged across publishers. The upper curve is the average weighted by the publisher's view-hours. The weighted average is always higher, indicating larger publishers tend to use more protocols. Despite fluctuations, the average number of protocols has remained a bit below two, and the weighted average higher than two. This consistency is likely because the growth in DASH has coincided with the decline of HDS.

4.2 Device Playback

Understanding the set of user devices that a publisher supports is important because (i) implementing and maintaining video players
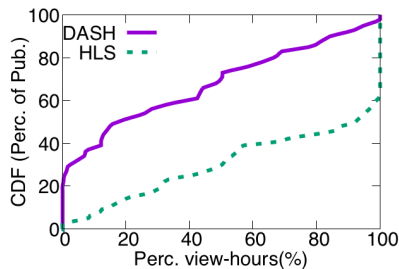
Figure 4: CDF across publishers of percentage of view-hours served via DASH and HLS in the latest snapshot.
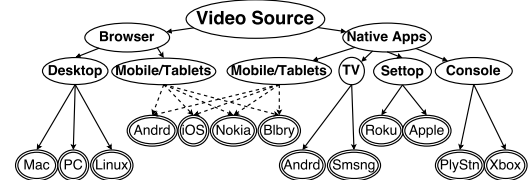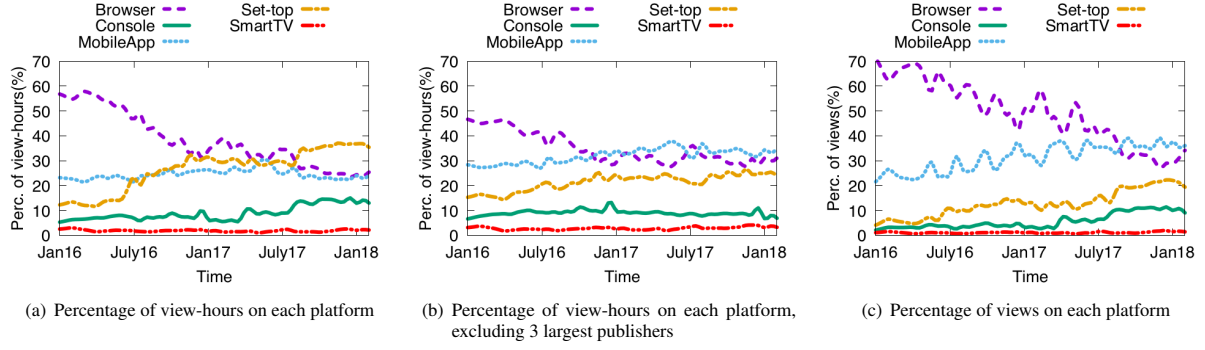

Figure 5: Target platforms for video publishers

for a range of platforms requires significant effort; and (ii) the popularity of a platform can impact the publisher's decision of whether to support it.

Prevalence by platform. Video is consumed (Fig. 5) on a variety of devices which can broadly be classified into two *platform types*: browsers and apps. Video is consumed on desktops, laptops, tablets and mobile devices using browsers on these devices. Video is also consumed using apps on mobile devices, smart TVs, streaming set-top boxes (we use the term set-top boxes to refer to streaming set-top boxes (§1)) and gaming consoles. We differentiate between set-top boxes and smart TVs for two reasons. First, we find that set-top boxes often require their own specific software development kits for application development [3, 7]. Second, set-top box usage is not necessarily limited to non-smart TVs. In fact set-top boxes may also be used with smart TVs, for example, to augment the range of available content, so it is important to distinguish between the two.

(a) Percentage of view-hours on each platform

(b) Percentage of view-hours on each platform, excluding 3 largest publishers
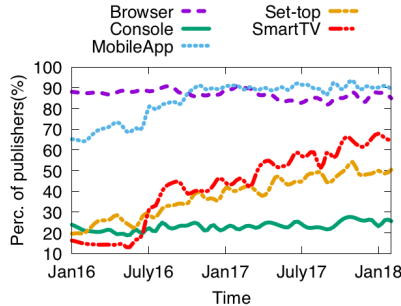
(c) Percentage of views on each platform

Figure 6: Over time, percentage of view-hours, view-hours excluding 3 largest publishers, and views on each type of platform

We now explore the prevalence of video consumption across these 4 app-based platform categories, and across browsers (which includes browser usage on mobile devices[6]).

Across Publishers. Fig. 7 shows that, over the 27 month period, support has grown most significantly for set-top boxes and smart TVs (from under 20% of publishers to above 50% and 60% of publishers respectively today). Complementary to this growth, it is interesting to note that YouTube, which has traditionally focused on browsers and mobile devices [8], also recently released an application to support large screen devices such as Samsung smart TV and Apple's tvOS set-top box [34]. This shows that video publishers in general are now paying great attention towards supporting large screen devices. Finally, we also observe that there has also been growth in mobile

Figure 7: Percentage of publishers supporting each platforms
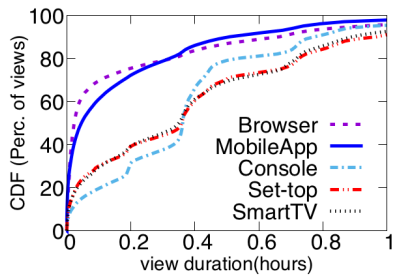
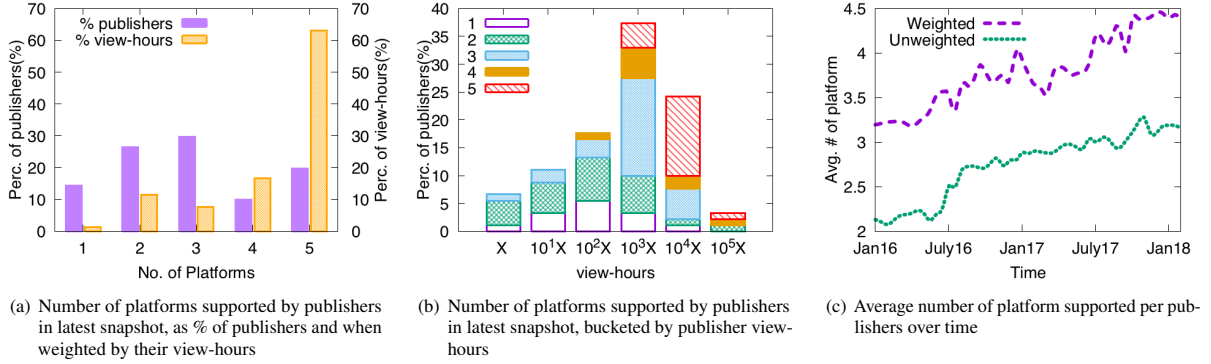**Figure 8: CDF of individual view duration for each platform**

applications, and, as expected, almost all publishers support browsers and mobile apps today.

By view-hours. Fig. 6(a) shows that, by percentage of view-hours served by different platforms, browser viewership has declined from nearly 60% to less than 25% today. Despite publishers aggressively increasing support for smart TVs during the last two years, their share of view-hours has stayed at less than 5%. Interestingly, the set-top category has grown the most, with the largest share of view-hours (nearly 40%) in the latest snapshot, while mobile app viewership has stayed steady at about 20-25%. These results are qualitatively consistent with a recent report [24] that shows that, among Netflix users, set-top boxes and smart TVs accounted for 70% of view-hours, while web browsers and mobiles accounted for 15% each in early 2018.

To understand whether large publishers bias our observations, Fig. 6(b) shows view-hour trends when we remove the top few publishers. There are some differences in trends, with mobile app viewing surpassing all other platforms over time, and set-top viewing growing at a slower rate. However, the results overall are qualitatively similar, indicating that platform usage trends in our dataset are *not* being driven by the largest publishers alone, unlike the trend with DASH adoption.

By views. The growth in view-hours with set-top boxes could be caused by longer view durations or by more views. To investigate this further, Fig. 6(c) depicts the fraction of *views* served across different platforms (including the three large publishers). While views with set-top boxes have grown to 20% of views in the latest snapshot, they lag behind the set-top view-hour growth (to nearly 40% in Fig. 6(a)). Taken together, Fig. 6(a) and Fig. 6(c) suggest that mobile app views are of shorter duration, while set-top views are of longer duration. Fig. 8 confirms this intuition, depicting the CDF of individual view duration (in hours, with the X-axis truncated at 1 hour) for each platform in our latest snapshot. Only 24% of mobile and browser views last longer than 0.2 hours, while more than 60% of set-top views last longer than 0.2 hours.

Trends within platforms. An examination of trends of device usage within each of the top 3 platforms also shows interesting trends, some well-known, others less so. Among browsers, the view-hours for HTML5 increased from about 25% to nearly 60% within the two year period (Fig. 10(a)). This increase came at the expense

---

[6]While smart TVs and set-top boxes support browsers, we see very little video consumption on these browsers.

(a) Number of platforms supported by publishers in latest snapshot, as % of publishers and when weighted by their view-hours

(b) Number of platforms supported by publishers in latest snapshot, bucketed by publisher view-hours

(c) Average number of platform supported per publishers over time

Figure 9: Number of platforms supported per publisher (by % of publishers and by their view-hours)

(a) Web browsers

(b) Mobiles/Tablets

(c) set-top boxes
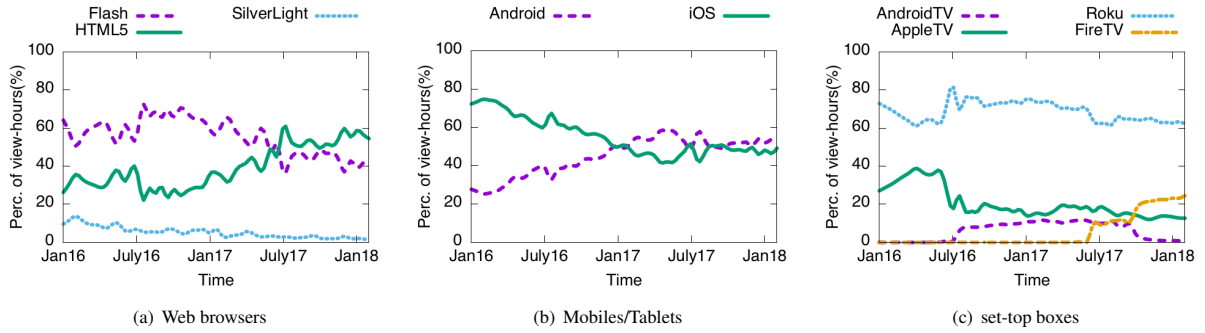
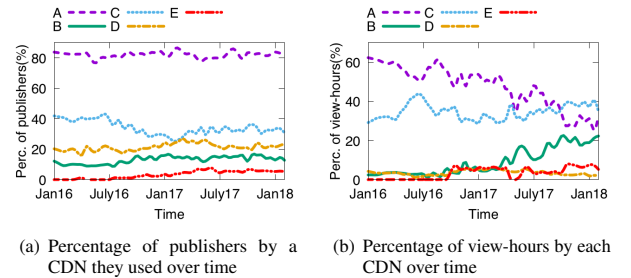**Figure 10: Percentage of view-hours served by specific devices belonging to the same platform.**

of a reduction of view-hours in other browser-based players, especially Flash. Among mobile devices, view-hours for Android devices have increased significantly (Fig. 10(b)), and both Android and iOS have comparable viewership in the latest snapshot. Finally, among set-top boxes (Fig. 10(c)) Roku devices are dominant in terms of view-hours, but AppleTV and FireTV account for a non-negligible percentage of view-hours. Overall, the results indicate that a publisher must not only cope with multiple platforms but also multiple devices within each platform, which can contribute to significant management complexity (§5).

Number of platforms per publisher. Fig. 9(a) characterizes the number of platforms supported by publishers. Over 85% of publishers support more than one platform, and over 95% of view-hours are attributable to these publishers. 30% of publishers support all 5 platforms and these publishers account for over 60% of the view-hours. As with other dimensions, the number of platforms supported increases with view-hours (Fig. 9(b)). For instance in the bucket corresponding to $10^3 X$ to $10^4 X$ view-hours, the vast majority of publishers support at least 3 platforms, and nearly half support all 5 platforms. Finally, Fig. 9(c) shows that the average and the view-hour weighted average of the number of platforms supported by publishers have increased by 48% and 37% respectively over the two year period. Publishers support more than 3 platforms on average in the latest snapshot, with the weighted average being nearly 4.5.
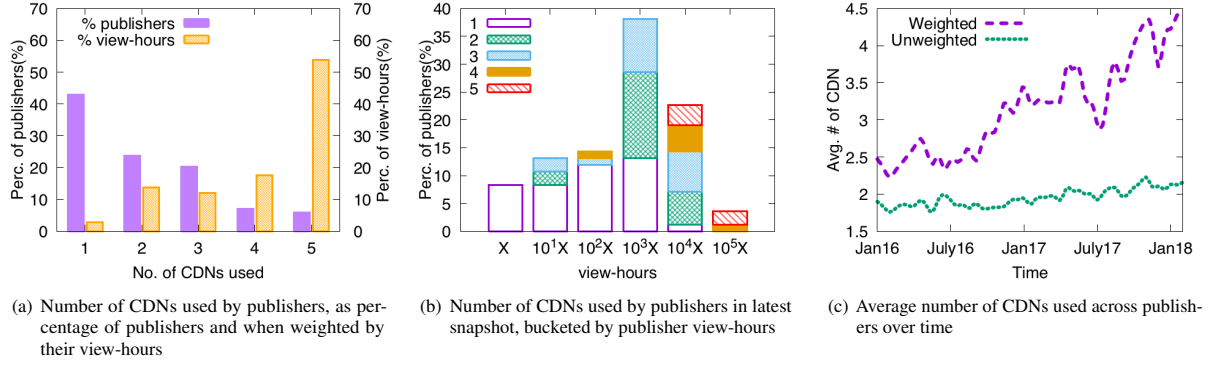
4.3 Content Distribution

Once the content is packaged, it is distributed to end users using content distribution networks (CDNs). CDNs work by situating the content closer to the end user. Understanding this dimension is important because CDN usage can have significant performance impact [61, 66, 69, 73]. Further, publishers can employ multiple CDNs which can lead to complexity in video management (§5).

Prevalence by CDN. Across all publishers we observed 36 different CDNs in our dataset. This list included both regional and international CDNs. Further, some publishers had their own internal CDNs (sometimes used in conjunction with external CDNs). Of these, over 93% of the view-hours were served by 5 CDNs, indicating that video

(a) Percentage of publishers by a CDN they used over time

(b) Percentage of view-hours by each CDN over time

Figure 11: Analysis of CDNs based on percentage of publishers and view-hours for past 27 months

(a) Number of CDNs used by publishers, as percentage of publishers and when weighted by their view-hours

(b) Number of CDNs used by publishers in latest snapshot, bucketed by publisher view-hours

(c) Average number of CDNs used across publishers over time

Figure 12: Number of CDNs used by publishers (by % of publishers and by their view-hours).

viewership is concentrated among a handful of CDNs. We analyze the opportunities arising from this consolidation in §6.

Across publishers. Fig. 11(a) shows the percentage of publishers across time that use each of the top 5 CDNs (anonymized). One CDN, A dominates, with nearly 80% of the publishers using it, while only 30% use the second most dominant CDN C. Longitudinally, these numbers have remained more or less steady.

By view-hours. Fig. 11(b) shows the percentage of the view-hours served by each of these CDNs. In the current snapshot, 3 CDNs (A, B and C) each account for 20-35% of the view-hours, while the other 2 account for about 5% or less each. Some CDNs use anycast to direct a client to a particular server [54], but anycast is susceptible to BGP route changes that sever ongoing TCP connections, raising concerns that anycast may not be suitable for large transfers [79]. We discovered that 1 of the top 3 CDNs in our dataset uses anycast, suggesting that anycast route instability has not been a blocking factor in the reliable delivery of video chunks.

While longitudinal CDN usage across publishers has largely remained stable, the longitudinal trend of view-hours shows significant changes. In particular, CDN A is no more dominant in our latest snapshot, where CDN B and C served a greater share of view-hours. Again, these results highlight the importance of considering view-hours in the analysis, an aspect not considered by prior work (§3).

Number of CDNs per publisher. We next characterize publishers by the number of CDNs they use.

Fig. 12(a) shows the percentage of publishers that use a given number of CDNs, and the percentage of view-hours that may be attributed to these publishers in our latest snapshot. While more than 40% of publishers use only a single CDN, they account for less than 5% of the view-hours. In contrast, less than 10% of publishers use 5 CDNs, but these publishers account for more than 50% of view-hours.

Fig. 12(b) shows the number of CDNs used by publishers classified by their view-hours. The results indicate that the percentage of publishers that use multiple CDNs increases with the number of view-hours attributable to the publisher. For example, at the extremes, all publishers with more than 10^5X view-hours use at least 4 CDNs, while all publishers with less than X daily view-hours use a single CDN. In the $10^3X - 10^4X$ bucket, the number of CDNs

used ranges from 1 to 3, while in the $10^4X - 10^5X$ bucket, the number of CDNs ranges from 1 to 5.

Finally, Fig. 12(c) shows the longitudinal trend for the average number of CDNs used by publishers, and the weighted average (weighted by the publisher's view-hours). While there is some growth in the average CDNs per publisher (with the average exceeding 2 in the latest snapshot), the weighted average grows much faster and is nearly 4.5 in the latest snapshot.

Live video has some different demands than video-on-demand (VoD), especially low end-to-end latency from video capture to viewing, and so we were interested in whether publishers favored particular CDNs for one type of content versus the other, perhaps due to different CDN features or latency. Of publishers which use multiple CDNs and serve both live and VoD traffic, 30% use at least one CDN only for VoD traffic, and 19% use at least one CDN only for live traffic. In one extreme case, a publisher used one CDN for all its VoD traffic and a different CDN for all its live video. However, most CDNs that were used exclusively for live content by one publisher were used exclusively for VoD content by another publisher. Thus, no CDN dominated others for live video, and our results seem to reflect opaque management plane decisions of publishers. We have left it to future work to explain the rationale for CDN choices amongst publishers.

4.4 Summary

Several common themes run through our analysis of these three dimensions of management complexity.

- In no dimension does a single alternative dominate in terms of view-hours. View-hours are roughly equal between HLS and DASH; across browser, mobile, and set-up; and across three large CDNs.
- More than 90% of view-hours can be attributed to publishers who support more than 1 protocol. The same is true of publishers who use more than 1 CDN, and publishers who support more than 1 platform.
- Publishers with more view-hours support more choices in each dimension. The average number of choices, weighted by view-hours, is 2.2 for protocols, 4.5 for CDNs and 4.5 for platforms.
- At least two of our trends (increase in DASH usage, and the emergence of 3 CDNs with comparable view-hours) are driven

by large publishers. However, large publishers alone do not drive trends in platform usage.

By assessing the distribution of view-hours, we observe new trends and provide additional insights on known trends:

- By view-hours, set-top box usage is significant, even exceeding browsers and mobile apps. This sharp rise of set-top box usage is not well documented and can drive the adoption of higher resolution video such as 4K video.
- Prior work has not quantified the distribution of multi-CDN usage. We find that almost 80% of view-hours are from publishers using 4 and 5 CDNs. While 2 or 3 CDNs are sufficient for resilience or load balancing, additional CDNs appear to be necessary for improved coverage.
- Given industry excitement with DASH [41], we expected to find significant DASH support among our publishers. While over 40% of our publishers support DASH, a small number of large publishers account for most of the DASH view-hours. In working with publishers, we have experienced quality issues with DASH implementations, so it might take some more time for the DASH ecosystem to mature to the point where small publishers also use more DASH.
- Our dataset shows negligible use of RTMP even though several of our publishers serve live content. RTMP provides low latency live streaming, but it has some scalability issues and lacks widespread device support.
- Other studies report high mobile view shares [22], and we find that these have indeed risen over time, but mobile app view-hours have not increased by a corresponding amount, because view durations on mobile devices tend to be small.
- Prior work has quantified the demise of Flash, reporting a 96% drop in Flash views for one browser [28]. We find a much more modest drop, with about 40% of browser view-hours attributable to Flash, down from 60% at the beginning of our study.

5 UNDERSTANDING MANAGEMENT COMPLEXITY

Our results in §4 have shown that publishers must deal with significant *diversity* across all components of the management plane. This diversity can impact the *complexity* of management tasks. In this section, we propose measures to capture this complexity, and explore how these measures correlate with publisher view-hours, an approximate indication of publisher size. A correlation indicates that management complexity is higher for large publishers and low for small publishers. If, however, even small publishers incur high management complexity, this may indicate a high barrier to entry since a publisher who targets modest viewership early in its business growth must still pay for high management costs.

Some examples of video management tasks include:

Software development and maintenance. Video publishers must build and maintain players for different devices and browsers. Typically, content publishers use device specific SDKs (released by device vendors, or third parties) [5, 21, 33]. A publisher may not only need to maintain multiple code bases corresponding to the different supported devices (§4), but may also need to support multiple versions

of the SDKs to support legacy devices. Besides the one-time development cost, there is an ongoing maintenance cost associated with rolling out new features, and fixing software bugs.

Packaging. For each video ID, a publisher needs to package the content for different streaming protocols. Packaging may be performed by the CDN or other third parties [2, 13, 27, 31]. Though packaging performed by CDNs may offer better economies of scale, the associated overheads remain irrespective of who does the packaging.

Failure Triaging. Troubleshooting video performance problems is challenging, and poor performance may be due to a CDN, the network or the user's device [62, 67], or a combination of these factors [67]. In addition, performance problems may be associated with a particular streaming protocol (*e.g.,* manifest files may have errors for specific protocols).

Measures of management complexity. Each of these tasks has an associated *complexity* that depends on the three components we study (protocols, CDNs, and devices). We list below *complexity measures* for video management, motivated in part by prior work on quantifying complexity in other domains such as web page complexity [51], and router configuration complexity [50]. Other measures are possible, and we have left an extended exploration to future work.

Management plane combinations. One measure of complexity is the number of unique combinations of CDN, streaming protocol, and the end user's device that a publisher supports. This relates to the complexity of triaging failures. A failure can be caused by one of the components (*e.g.,* CDN or protocol), an interaction between two components (*e.g.,* a specific CDN's implementation of HLS), or an interaction across all three components (*e.g.,* we have observed a failure caused by the interaction between a Chromecast implementation using SmoothStreaming on a specific CDN). In the worst case, it might be necessary to examine all combinations to triage a failure. Indeed, Conviva, triages failures automatically by aggregating failure reports across all management plane combinations [10]. More broadly, failure triaging may also depend on other factors such as the choice of ISP which we do not consider in this paper.

Protocol-titles. The product of protocols used by a publisher and the number of unique video IDs (or video titles) for a publisher captures the packaging costs for the publisher's content[7]. Intuitively, each publisher has to package each video separately for each protocol. This measure determines the compute and storage resources needed to package the publisher's contents and can impact the lag experienced by users for live content.

Unique SDKs. Defined as the number of unique versions of SDKs and browsers supported by a publisher across all devices, this measure captures the software development and maintenance complexity. The metric may also relate to the complexity of triaging a failure related to the device, if the failure is specific to an SDK version or browser.

Correlation between management complexity and publisher view-hours. Fig. 13(a) presents a scatter plot that shows how the *management plane combinations* metric (on a log scale) correlates with the view-hours (on a log-scale) served by the publisher. We

---

[7]To a first approximation. Packaging cost may also depend on the *length* of each video, which we do not have.

(a) Management plane combinations

(b) Protocol-titles

(c) Unique SDKs
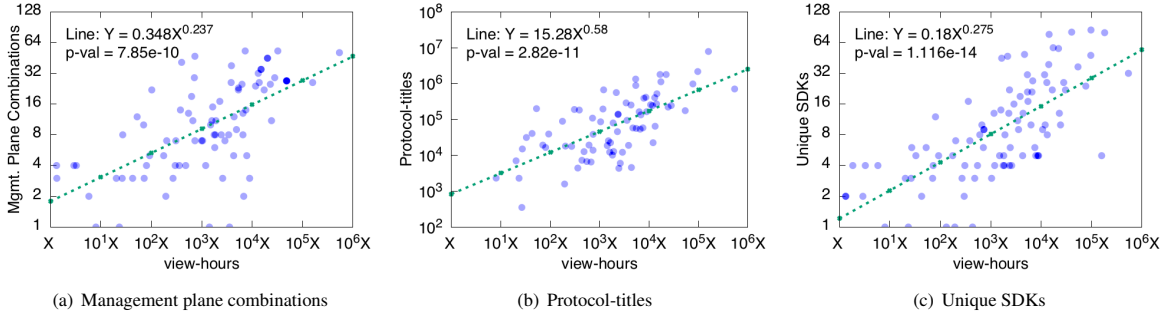
Figure 13: Correlation between different measures of complexity and publisher view-hours
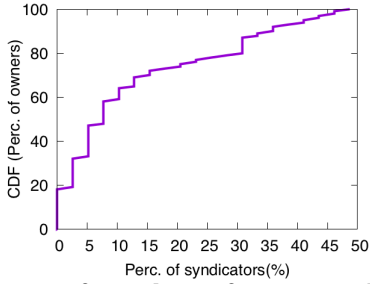
Figure 14: Degree of prevalence of content syndication: some content owners syndicate to nearly half the full syndicators.

also add a line of best fit using linear regression. The slope of the line shows that when the view-hours increase by a factor of 10, the number of management plane combinations increases by a factor of $1.72\times$, indicating a sub-linear growth in complexity with publisher size.

Fig. 13(b) and Fig. 13(c) respectively show similar scatter plots and lines of best fit for both the *Protocol-titles* metric, and the *Unique SDKs* metric. Again, both graphs indicate that the complexity measures increase sub-linearly with publisher view-hours: when view-hours increase by an order of magnitude the *Protocol-titles* grows by $3.8\times$, while the *Unique SDKs* metric grows by $1.8\times$ with the biggest publishers having to maintain up to 85 different code bases.

In each case, the linear fit is statistically significant, with p-values at the 0.05 level of significance smaller than 10^{-9}.

6 MANAGEMENT OF SYNDICATION

We next explore how today's structure of management planes, where each publisher makes independent decisions on the choice of protocols, CDNs and playback devices, can result in sub-optimal performance when content is *syndicated*. *Syndicators* license and serve content obtained from multiple *content owners*. Conversely, content owners may distribute their content through multiple syndicators.

The prevalence of syndication. For each publisher-ID-video-ID pair, a flag specifies whether the content is owned or syndicated. With this, we can determine, for a given content owner, what fraction of full syndicators have syndicated that owner's content. Fig. 14 shows the CDF of the percentage of syndicators used by each content owner. The figure shows that syndication is prevalent – more than 80% of content owners use at least one syndicator, and 20% of content

owners syndicate their content to almost $\frac{1}{3}$rd of all full syndicators. These numbers are conservative, since we do not have coverage of all the syndicators, and since we have not considered partial syndication. Overall, these numbers suggest that content syndication is significant in online video delivery.

Incorporating syndication in management planes. Today, because each publisher runs an independent management plane instance, the easiest way to syndicate content is that the content owner provides a master or "mezzanine" copy of the content to each of its syndicators which then packages and distributes the content through its video management plane.

In this *independent syndication* model, sub-optimal outcomes might result because syndicators can make independent decisions on video packaging choices. In this paper, we illustrate two such outcomes to motivate why it is important to study video management planes: (a) different performance for the same syndicated content resulting from different bitrate choices; (b) redundancy in CDN storage usage because multiple copies of the same content can be stored on a CDN using different encodings or protocols.

To quantify these, we focus on a popular video catalogue that has 10 syndicators. We focus on a single video ID from the catalogue because the bitrates used to encode different videos may vary based on the content [38].

Bitrate choices for syndicated content. As described before, bitrate choices decide, for each platform, the resolutions and qualities at which the video is available. With independent syndication, an owner and a syndicator can make different bitrate choices for the same video ID. For each video ID there is associated information about the audio and video bitrates used in the encoding, as well as other information such as the duration of each chunk. In general, each publisher may have different set of bitrates for the same video ID based on factors such as the streaming protocol, type of device,

(a) On ISP X, CDN A

(b) On ISP Y, CDN B
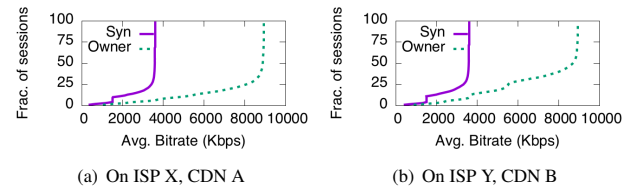
Figure 15: Average bitrate performance of California based iPad clients of owner and of syndicator across different ISPs and CDNs.

(a) On ISP X, CDN A   (b) On ISP Y, CDN B
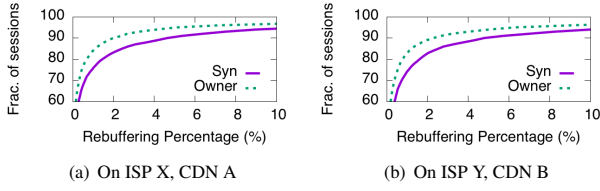
Figure 16: Rebuffering performance of California based iPad clients of owner and of syndicator across different ISPs and CDNs.

and network connectivity (WiFi, 4G, Wired). For fair comparison, we compare the bitrates served to the same type of device over the same type of Internet connection (WiFi, 4G, Wired).

Bitrate choices can vary widely. Fig. 17 shows the bitrates used by the syndicators (S1 to S10), as well as the bitrates offered by the original content owner (O) for iPad devices over a WiFi network. The figure shows a significant difference both in the number of bitrates, the range of bitrates, and individual bitrate choices. At one extreme, S2 encodes the video into only 3 different bitrates, while at the other end, S9 employs 14 bitrates. The owner uses 9 different bitrates and offers a bitrate that exceeds 8192 Kbps, while the highest bitrate offered by S1 is 7x lower and only a little above 1024 Kbps. We have also performed a similar comparison for other device types, and observed similar heterogeneity in bitrate decisions made by the content owner and various syndicators.

Bitrate choices can impact performance. We also study the *performance* achieved by clients of some of these syndicators. Two widely used measures of video delivery performance are the average bitrate of each view, and the rebuffering ratio (the fraction of the view that experiences rebuffering) [49, 57].

Fig. 15 shows the distribution of average bitrates observed by iPad clients of a syndicator (S7 in Fig. 17) and the owner, for our selected video ID, across two different ISP/CDN combinations in March 2018. Further, we restricted the geo-location of clients to California, USA. Consistently, the content owner's clients get much better average bitrates: at the median, the average bitrate of the owner's clients is $2.5\times$ that of the syndicator. Interestingly, clients of the owner also perceive lower rebuffering ratios (Fig. 16), with almost 40% lower rebuffering at the 90-th percentile. We observed similar results for other device, ISP and CDN combinations.
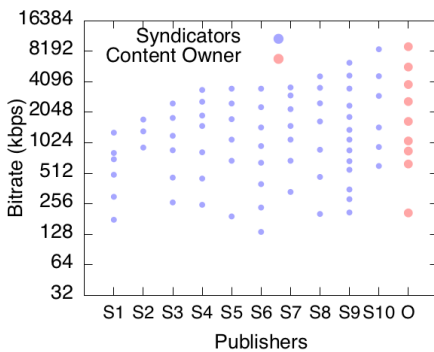
Figure 17: Bitrate selection decisions for a video ID from a popular catalogue by the owner and ten syndicators.
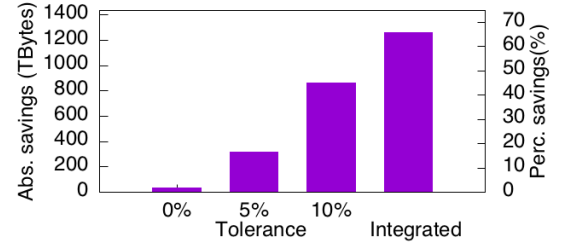
Figure 18: Storage savings under different syndication models for content served by an owner and two syndicators.

While we cannot comprehensively answer why syndicators select widely varying bitrates, our hypothesis is that the owner selected its bitrates to provide better experience for its users, while the syndicator's choices were dictated in part by the increased storage and encoding costs required for higher bitrates.

Redundancy in CDN storage. Independent syndication can also result in redundant storage in CDNs. In this section, we explore this for a popular video catalogue syndicated by two syndicators from the owner. In this case, the owner stores the catalogue on two CDNs A and B, and uses 9 bitrates. One of its syndicators stores the same catalogue on 3 CDNs A, B and C, but encodes the videos using 7 bitrates. Another syndicator stores the catalogue on A, B and another CDN D, but encodes the videos using 14 different bitrates. These different management choices for the same content arise largely because of the independent syndication model.

We focus on a setting where publishers proactively push video content to a CDN origin server which serves cache misses from CDN edge servers [64]. This setting is commonly used in practice, especially for popular video content. We quantify the redundancy in storage in CDN origin servers. While there is likely some redundancy in edge servers as well, this is harder to quantify as that depends on content access patterns.

Quantifying storage redundancy with independent syndication. To quantify redundant storage, we first compute the total storage required for the entire catalogue. We do this by (i) multiplying for each video ID, its encoded bitrates by its duration in seconds, and summing these products to obtain storage per video ID and (ii) by summing the storage per video ID across all videos in the catalogue to obtain the total catalogue storage. This results in a total storage requirement of 1916 TB across the 3 publishers (content owner and two syndicators) for each of the common CDNs (A and B).

We next explore the storage savings achievable for this catalogue if a CDN removes redundant copies of chunks with the same, or similar bitrates (those within a small *tolerance* factor). This is motivated by the observation that the syndicators and content owners often have similar or identical bitrates (Fig. 17). This occurs in practice because, even though publishers make independent bitrate choices, they tend to follow guidelines recommended by streaming protocol specifications. For example, the HLS specifications recommend that publishers make available at least one bitrate under 192 Kbps and that each successive bitrate be within a multiplicative factor of 1.5-2× of the previous [6].

Fig. 18 (left three bars) shows the absolute and percentage storage savings with the above model. Even with a 5% tolerance, CDNs A

and B can save 316.1 TBs (16.5%) each, and at 10%, they can each save 865 TBs (45.2%).

Integrated syndication. Another form of syndication is what can be called *integrated syndication*, in which the owner's content delivery mechanism is integrated into the syndicator's playback software. There are two variants of integrated syndication: (i) *API integration*, where the syndicator uses the owner's manifest file and CDN; and (ii) *app integration*, where the syndicator embeds the owner's app into its own.

The rightmost bar of Fig. 18 shows that with integrated syndication, A and B each save 1257 TB (65.6%). In addition, especially with app integration, syndicators cannot choose different bitrates than content owners, so performance differences similar to Fig. 15 are unlikely to arise.

While integrated syndication has potential, many logistical challenges must be addressed to make it a reality. For instance, with app integration, syndicators have to integrate apps for every owner they syndicate from. While API integration is potentially logistically easier, accounting mechanisms must be developed to distinguish CDN usage by clients of the syndicator and the owner. Future work should explore better ways to improve the management of syndication.

7   RELATED WORK

Characterizing Video Services. YouTube and Netflix alone have been the subject of numerous studies over the years [45, 46, 52, 53, 55, 56, 59, 63, 77, 81, 82]. This body of work has studied several aspects, including 1) architecture, serving strategy and its evolution, 2) characterization of videos in terms of encoded bitrates, total number of videos, popularity, caching, and 3) the user access patterns and quality of experience *etc.* Ghasemi *et al.* [62] conducted an in-depth study of Yahoo's video serving infrastructure to reveal problems in different points in the video delivery pipeline. Other work has also examined different types of video services including a Pay-TV [44], cellular video [58], an on-demand service [70] and user-generated live streaming services [75, 78]. These papers focus on one or a handful of online publishers, but our work focuses on characterizing management plane practices across a large number of online video publishers.

Industry surveys. Because of the growing interest in Internet video, several industry surveys [13, 22, 37, 41] have examined the video ecosystem. A 2017 industry study by Bitmovin surveyed 380 video developers (individuals or companies associated with the Internet video business in various ways). This study characterized streaming protocols, encoding formats, devices, DRM *etc.*. An earlier 2016 study [13] characterized aggregate distributions across many of the same dimensions as [41]. Another prior industry survey [18] and an anecdotal report [26] discuss the percentage of publishers that use multiple CDNs (but do not discuss the number of CDNs used, or the fact different CDNs may be used for live and VoD). While valuable, none of these reports weigh findings by view-hours, or present trends across publishers categorized by view-hours, or present longitudinal analyses, as we do. Both of these methodological differences result in new findings and add insights to known trends. Finally, while [22] does presents some trends with respect to device usage, other dimensions are not considered. In addition, we go

beyond all these reports by considering the implications of these trends for management complexity, and syndicated content.

Quantifying diversity and complexity. Prior work has captured diversity of mobile users [60] and apps [74] and the complexity of web pages [51] and routers [50]. While we draw inspiration from these works, our focus is on video management planes, a different domain.

8   CONCLUSION

The Internet video management plane, which is responsible for packaging video content and for ensuring playback across different devices, has received relatively little research attention. Using data collected by Conviva, a streaming TV measurement and intelligence platform, over a period of two years from over one hundred video publishers, we find that there exists significant diversity across the three aspects of video management we study (packaging, CDN usage, and playback device usage): large publishers support 3-4 protocols, 5 CDNs and 5 different device types. This diversity adds complexity to several management tasks such as failure triaging, software management, and encoding. We find that complexity metrics for these tasks are sub-linearly related to the number of view-hours. Finally, the structure of today's management planes can lead to variable delivery performance for syndicated content. Integrating management planes for syndicated content can avoid this as well as reduce CDN origin server storage requirements, and future work can explore mechanisms for integrated syndication, as well as analyze new complexity metrics, and approaches to cope with diversity and reduce management complexity.

REFERENCES

[1] Adobe: Adobe HTTP Dynamic Streaming. www.adobe.com/products/hds-dynamic-streaming.html.
[2] Amazon AWS Media Package. https://aws.amazon.com/mediapackage/.
[3] Amazon: Fire App Builder. https://developer.amazon.com/docs/fire-app-builder/download-and-build.html.
[4] Apple: Apple's HTTP Live Streaming. https://developer.apple.com/streaming/.
[5] Apple: AVFoundation framework . https://developer.apple.com/av-foundation/.
[6] Apple: Technical Note 2224 for HLS Streaming. https://developer.apple.com/library/content/technotes/tn2224/_index.html.
[7] Apple: tvOS. https://developer.apple.com/tvos/.
[8] comScore: Unlocking Mobile Measurement for YouTube in the U.S. https://www.comscore.com/Insights/Data-Mine/Unlocking-Mobile-Measurement-for-YouTube-in-the-US.
[9] Conviva. https://www.conviva.com/.
[10] Conviva: Precision Delivery Intelligence. https://www.conviva.com/whitepapers/.
[11] Conviva: The 2017 OTT Streaming Market Year in Review. https://www.conviva.com/blog/2017-ott-streaming-market-year-review/.
[12] DASH-IF: MPEG-DASH. https://mpeg.chiariglione.org/standards/mpeg-dash.
[13] encoding.com. http://1yy04i3k9fyt3vqjsf2mv610yvm-wpengine.netdna-ssl.com/files/2017-Global-Media-Formats-Report.pdf.
[14] Facebook Live. https://live.fb.com/.
[15] ITU: H.264. https://www.itu.int/rec/T-REC-H.264.
[16] ITU: H.265. https://www.itu.int/rec/T-REC-H.265.
[17] JW Player. https://www.jwplayer.com/.
[18] Level2: Over the top video delivery. http://www.level3.com/~/media/files/white-paper/en_cdn_wp_ovrtopvddlvry.ashx.

[19] Microsoft: Microsoft Smooth Streaming. http://www.iis.net/downloads/microsoft/smooth-streaming.

[20] Netflix: 2017 on Netflix - A Year in Bingeing. https://media.netflix.com/en/press-releases/2017-on-netflix-a-year-in-bingeingl.

[21] Nexplayer: Nexplayer Software Development Kit. https://nexplayersdk.com/.

[22] Ooyala: Global Video Index. http://go.ooyala.com/rs/447-EQK-225/images/Ooyala-Global-Video-Index-Q4-2017.pdf.

[23] Recode: Facebook Says Video Is Huge – 100-Million-Hours-Per-Day Huge. https://www.recode.net/2016/1/27/11589140/facebook-says-video-is-huge-100-million-hours-per-day-huge.

[24] Recode: You can watch Netflix on any screen you want, but you're probably watching it on a TV. https://www.recode.net/2018/3/7/17094610/netflix-70-percent-tv-viewing-statistics.

[25] Streaming Learning Center: DASH or HLS? Which is the best format today? https://streaminglearningcenter.com/blogs/dash-or-hls-which-is-the-best-format-today.html.

[26] Streamingmedia: Video: The Pros and Cons of a Multi-CDN Strategy. http://www.streamingmedia.com/Articles/Editorial/Short-Cuts/Video-The-Pros-and-Cons-of-a-Multi-CDN-Strategy-112351.aspx.

[27] Telestream. http://www.telestream.net/.

[28] The Chromium Projects: Flash Usage Trends. https://www.chromium.org/flash-roadmap/flash-usage-trends.

[29] The Wall Street Journal: YouTube Tops 1 Billion Hours of Video a Day, on Pace to Eclipse TV. https://www.wsj.com/articles/youtube-tops-1-billion-hours-of-video-a-day-on-pace-to-eclipse-tv-1488220851.

[30] Theoplayer. https://www.theoplayer.com/.

[31] Unified Streaming. http://www.unified-streaming.com.

[32] WebM: VP9. https://www.webmproject.org/vp9/.

[33] Xbox: XDK Software Development Kit. https://www.xbox.com/en-US/developers.

[34] YouTube Official Blog: Lean back and watch YouTube TV with a new app for your big screen. https://youtube.googleblog.com/2017/10/lean-back-and-watch-youtube-tv-with-new.html.

[35] YouTube: You know what's cool? A billion hours. https://youtube.googleblog.com/2017/02/you-know-whats-cool-billion-hours.html.

[36] Cisco: It Came to Me in a Stream... : https://www.cisco.com/web/about/ac79/docs/sp/Online-Video-Consumption_Consumers.pdf, 2012.

[37] DASH-IF: Survey of European Broadcaster on MPEG-DASH. http://dashif.org/wp-content/uploads/2015/04/Survey-of-the-European-Broadcasters-on-MPEG-DASH-Whitepaper-V2.1.pdf, 2013.

[38] Netflix: Per-title Encode Optimization, Dec 2015. https://medium.com/netflix-techblog/per-title-encode-optimization-7e99442b62a2.

[39] Sandvine: Global Internet phenomena report . https://www.sandvine.com/trends/global-internet-phenomena/, 2015.

[40] Apple: fMP4 support on Apple devices. https://developer.apple.com/streaming/examples/, 2016.

[41] Bitmovin: Video Developer Survey. https://bitmovin.com/whitepapers/Bitmovin-Developer-Survey.pdf, Sept. 2017.

[42] Cisco: Visual Networking Index: Global Mobile Data Traffic Forecast Update 2016-2021 . http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html, 2017.

[43] comScore: OTT breaks out of its Netflix shell . https://www.comscore.com/Insights/Blog/OTT-Breaks-Out-of-Its-Netflix-Shell, 2017.

[44] H. Abrahamsson and M. Nordmark. Program Popularity and Viewer Behaviour in a Large TV-on-demand System. In *Proceedings of the 2012 Internet Measurement Conference*, IMC '12, 2012.

[45] V. K. Adhikari, Y. Guo, F. Hao, V. Hilt, Z.-L. Zhang, M. Varvello, and M. Steiner. Measurement Study of Netflix, Hulu, and a Tale of Three CDNs. *IEEE/ACM Trans. Netw.*, 23(6), Dec. 2015.

[46] V. K. Adhikari, S. Jain, and Z.-L. Zhang. YouTube Traffic Dynamics and Its Interplay with a Tier-1 ISP: An ISP Perspective. In *Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement*, IMC '10, 2010.

[47] S. Akhshabi, A. C. Begen, and C. Dovrolis. An Experimental Evaluation of Rate-adaptation Algorithms in Adaptive Streaming over HTTP. In *Proceedings of the Second Annual ACM Conference on Multimedia Systems*, MMSys '11, 2011.

[48] Z. Akhtar, Y. S. Nam, R. Govindan, S. Rao, J. Chen, E. Katz-Bassett, B. M. Ribeiro, J. Zhan, and H. Zhang. Oboe:Auto-tuning video ABR algorithms to network conditions. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*, SIGCOMM '18, 2018.

[49] Balachandran, Athula and Sekar, Vyas and Akella, Aditya and Seshan, Srinivasan and Stoica, Ion and Zhang, Hui. Developing a predictive model of quality of experience for internet video. 2013.

[50] T. Benson, A. Akella, and D. Maltz. Unraveling the Complexity of Network Management. In *Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation*, NSDI'09, 2009.

[51] M. Butkiewicz, H. V. Madhyastha, and V. Sekar. Understanding Website Complexity: Measurements, Metrics, and Implications. In *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, IMC '11, 2011.

[52] T. Böttger, F. Cuadrado, G. Tyson, I. Castro, and S. Uhlig. A Hypergiant's View of Internet. In *SIGCOMM Computer Communication Review*, CCR '18, 2018.

[53] M. Calder, X. Fan, Z. Hu, E. Katz-Bassett, J. Heidemann, and R. Govindan. Mapping the Expansion of Google's Serving Infrastructure. In *Proceedings of the 2013 Conference on Internet Measurement Conference*, IMC '13, 2013.

[54] M. Calder, A. Flavel, E. Katz-Bassett, R. Mahajan, and J. Padhye. Analyzing the Performance of an Anycast CDN. In *Proceedings of the 2015 Internet Measurement Conference*, IMC '15, 2015.

[55] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon. I Tube, You Tube, Everybody Tubes: Analyzing the World's Largest User Generated Content Video System. In *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement*, IMC '07, 2007.

[56] Y. Ding, Y. Du, Y. Hu, Z. Liu, L. Wang, K. Ross, and A. Ghose. Broadcast Yourself: Understanding YouTube Uploaders. In *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, IMC '11, 2011.

[57] F. Dobrian, V. Sekar, A. Awan, I. Stoica, D. Joseph, A. Ganjam, J. Zhan, and H. Zhang. Understanding the Impact of Video Quality on User Engagement. In *Proceedings of the ACM SIGCOMM 2011 Conference*, SIGCOMM '11, 2011.

[58] J. Erman, A. Gerber, K. K. Ramadrishnan, S. Sen, and O. Spatscheck. Over the Top Video: The Gorilla in Cellular Networks. In *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, IMC '11, 2011.

[59] A. Finamore, M. Mellia, M. M. Munafò, R. Torres, and S. G. Rao. YouTube Everywhere: Impact of Device and Infrastructure Synergies on User Experience. In *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, IMC '11, 2011.

[60] K. Fukuda, H. Asai, and K. Nagami. Tracking the Evolution and Diversity in Network Usage of Smartphones. In *Proceedings of the 2015 Internet Measurement Conference*, IMC '15, 2015.

[61] A. Ganjam, F. Siddiqui, J. Zhan, X. Liu, I. Stoica, J. Jiang, V. Sekar, and H. Zhang. C3: Internet-Scale Control Plane for Video Quality Optimization. In *12th USENIX Symposium on Networked Systems Design and Implementation, NSDI 15*, 2015.

[62] M. Ghasemi, P. Kanuparthy, A. Mansy, T. Benson, and J. Rexford. Performance Characterization of a Commercial Video Streaming Service. In *Proceedings of the 2016 ACM on Internet Measurement Conference*, IMC '16, 2016.

[63] P. Gill, M. Arlitt, Z. Li, and A. Mahanti. YouTube Traffic Characterization: A View from the Edge. In *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement*, IMC '07, 2007.

[64] Q. Huang, K. Birman, R. van Renesse, W. Lloyd, S. Kumar, and H. C. Li. An Analysis of Facebook Photo Caching. In *Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles*, SOSP '13, 2013.

[65] T.-Y. Huang, R. Johari, N. McKeown, M. Trunnell, and M. Watson. A Buffer-based Approach to Rate Adaptation: Evidence from a Large Video Streaming Service. In *Proceedings of the 2014 ACM Conference on SIGCOMM*, SIGCOMM '14, 2014.

[66] J. Jiang, V. Sekar, H. Milner, D. Shepherd, I. Stoica, and H. Zhang. CFA: A Practical Prediction System for Video QoE Optimization. In *13th USENIX Symposium on Networked Systems Design and Implementation, NSDI 16*, 2016.

[67] J. Jiang, V. Sekar, I. Stoica, and H. Zhang. Shedding Light on the Structure of Internet Video Quality Problems in the Wild. In *Proceedings of the Ninth ACM Conference on Emerging Networking Experiments and Technologies*, CoNEXT '13, 2013.

[68] J. Jiang, V. Sekar, and H. Zhang. Improving Fairness, Efficiency, and Stability in HTTP-based Adaptive Video Streaming with FESTIVE. In *Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies*, CoNEXT '12, 2012.

[69] J. Jiang, S. Sun, V. Sekar, and H. Zhang. Pytheas: Enabling Data-Driven Quality of Experience Optimization Using Group-Based Exploration-Exploitation. In *14th USENIX Symposium on Networked Systems Design and Implementation, NSDI 2017, Boston, MA, USA, March 27-29, '17*, 2017.

[70] Z. Li, J. Lin, M.-I. Akodjenou, G. Xie, M. A. Kaafar, Y. Jin, and G. Peng. Watching Videos from Everywhere: A Study of the PPTV Mobile VoD System. In *Proceedings of the 2012 Internet Measurement Conference*, IMC '12, 2012.

[71] H. Mao, R. Netravali, and M. Alizadeh. Neural Adaptive Video Streaming with Pensieve. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*, SIGCOMM '17, 2017.

[72] M. K. Mukerjee, I. N. Bozkurt, B. Maggs, S. Seshan, and H. Zhang. The Impact of Brokers on the Future of Content Delivery. In *Proceedings of the 15th ACM Workshop on Hot Topics in Networks*, HotNets '16, 2016.

[73] M. K. Mukerjee, D. Naylor, J. Jiang, D. Han, S. Seshan, and H. Zhang. Practical, Real-time Centralized Control for CDN-based Live Video Delivery. In *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, SIGCOMM '15, 2015.

[74] T. Petsas, A. Papadogiannakis, M. Polychronakis, E. P. Markatos, and T. Karagiannis. Rise of the Planet of the Apps: A Systematic Study of the Mobile App Ecosystem. In *Proceedings of the 2013 Conference on Internet Measurement Conference*, IMC '13, 2013.

[75] M. Siekkinen, E. Masala, and T. Kämäräinen. A First Look at Quality of Mobile Live Streaming Experience: The Case of Periscope. In *Proceedings of the 2016*

Internet Measurement Conference, IMC '16, 2016.

[76] K. Spiteri, R. Urgaonkar, and R. K. Sitaraman. BOLA: Near-Optimal Bitrate Adaptation for Online Videos. *CoRR*, 2016.

[77] R. Torres, A. Finamore, J. R. Kim, M. Mellia, M. M. Munafo, and S. Rao. Dissecting Video Server Selection Strategies in the YouTube CDN. In *Proceedings of the 2011 31st International Conference on Distributed Computing Systems*, ICDCS '11, 2011.

[78] B. Wang, X. Zhang, G. Wang, H. Zheng, and B. Y. Zhao. Anatomy of a Personalized Livestreaming System. In *Proceedings of the 2016 Internet Measurement Conference*, IMC '16, 2016.

[79] L. Wei and J. Heidemann. Does Anycast Hang up on You? In *IEEE International Workshop on Traffic Monitoring and Analysis*, Dublin, Ireland, 2017.

[80] X. Yin, A. Jindal, V. Sekar, and B. Sinopoli. A Control-Theoretic Approach for Dynamic Adaptive Video Streaming over HTTP. In *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, SIGCOMM '15, London, United Kingdom, 2015.

[81] J. Zhou, Y. Li, V. K. Adhikari, and Z.-L. Zhang. Counting YouTube Videos via Random Prefix Sampling. In *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, IMC '11, 2011.

[82] M. Zink, K. Suh, Y. Gu, and J. Kurose. Characteristics of YouTube Network Traffic at a Campus Network - Measurements, Models, and Implications. *Comput. Netw.*, Mar. 2009.