White Paper

Low Latency Streaming

The New Normal

thetvplatform.zattoo.com

Abstract

The White Paper also describes what video latency is, how

and from where it is introduced in the delivery chain, how the various distribution mechanisms of video differ when it comes to latency and, ultimately, what the stages are in the delivery chain where reduction in latency is most viable.

In describing Zattoo's solution, the White Paper explains the overall goals of the project, the considerations, trade-offs and decisions taken. Lastly, the paper explains the necessity of such a solution for service providers such as Zattoo and other IP based unicast services as the market continues to trend towards IP delivery dominating.

Understanding Video Latency

The term “latency” has been widely adopted by the TV industry merely for the purposes of defining that a delay exists when consuming live TV over an IP network. However, it is important to note that video latency bears no relation to network latency. Network latency is the time it takes in milliseconds for an IP packet to be transmitted from a sender to a receiver. Reasonably performant networks in the EU can be expected to transmit IP packets in under 20 milliseconds within a country and under 60 milliseconds between countries. To put this in perspective, it takes an average of 550 milliseconds for a live TV signal frame to be uplinked to and downlinked from a satellite.

So if the delivery mechanism for an IP consumer is actually faster than that of a satellite consumer, why does the IP consumer experience more of a delay?

One reason is that in order to deliver live TV over an IP network, a number of pre-processing activities need to be performed before transmission. Key is the need to process the encoded live TV signal from the broadcaster into a constant bit rate (CBR) profile. CBR is an essential ingredient for transmission on an IP network as it enables the internet service provider (ISP)/the network operator, to sufficiently prioritise and proportion its network to handle the delivery of live TV. In order to process the live TV channel into a CBR profile, the channel may first need to be decoded, and then re-encoded into the CBR profile. Analogous to re-packing an hastily packed suitcase before going on holiday and repacking it to make sure there’s at least 25% empty space to add stuff you bought on holiday . . . that takes time.

Additionally, the streaming service may be complemented by an adaptive bitrate feature. Adaptive bitrate (ABR) is where a number of lower quality CBR profiles of the channel are created to enable the consumers’ device to change the profile of the live TV channel to allow for fluctuations of the available IP network connection speed. On the video platform side, ABR is generally achieved by taking the originally encoded CBR profile of the content, which is generally the highest quality profile, and transcoding it to multiple lower quality CBR profiles. The transcoding process is commonly performed sequentially meaning that each of the lower quality CBR profiles are transcoded at the same time. Today, ABR is considered a must-have for any TV service

targeting streaming-capable devices. It is also particularly useful for implementing permission based business logic, such as enabling premium access to the highest resolutions and qualities. Other uses would be providing a fallback to lower quality profile where device and DRM requirements cannot be met or providing efficiency for more demanding or more expensive access networks such as mobile networks. In the suitcase analogy, ABR would be the equivalent of having a set of suitcases of various sizes so you can choose the size that best suits your needs, i.e. length of trip, space restrictions on your mode of transport, practicality and, indeed, the availability of the desired suitcase.

Next, the CBR stream needs to be fragmented into chunks of equal length and then encrypted with digital rights management (DRM) before transmission as IP packets. IP networks require the content to be encapsulated in an IP packet, the format of which is dependant on the client requesting the packet i.e. TCP/IP or UDP. In packaging the live TV channel in a supported structure, ISP’s can quickly and effectively switch and route the packet to the intended sender. Again to the suitcase analogy, this time you are a member of a family of five and you each have the same size of suitcase. First of all, you would not

attempt to pack all belongings into one case, but instead pack what can fit into each case. At check-in, you cannot pile all five suitcases on top of each other as they simply would not fit through the security scanner. Instead, you would need to place each case on the check-in belt for it to be tagged with your flight number and destination. Once tagged, it is put on the correct flight and arrives at your destination. Again, each of these minor steps takes time.

Moving towards the consumer side, another reason for the delay is what happens on the consumer's end device. When the IP packets have downloaded they have to be ordered, unpackaged, decrypted and loaded into the device's mediaplayer video-playout buffer. The media player is a piece of software within the application delivered by the network, or TV service operator, that enables the playout of video content on the device. Its primary role is to initiate the downloading of IP packets containing the required video content fragments using a manifest file which details the source of the IP packets and the various quality profiles the video content has available. The media player is also responsible for performing frequent bandwidth checks, commonly referred to as "estimates", to enable it to decide which quality profile is best suited based on the latest estimate. In order to enable the media player to seamlessly download and adapt video content quality profiles, it uses the video-playout buffer to store a defined number of seconds / fragments of video content before being played out to the consumer. As buffering is essential to providing a smooth and consistent experience, especially where an IP network is used as the delivery mechanism, it also adds more seconds to the overall delay. Returning for the last time to the suitcase analogy, buffering would be the foresight to pack enough items to last the entire trip plus one day in the event an overstay is required.

So now that we understand the various factors that add latency, we turn our focus to the two types of IP video stream transmission employed today, namely unicast and multicast, to see how these latency factors impact each.

Firstly, a quick introduction to both transmission methods:

(IP) Multicast is a connectionless, one-to-many transmission method built on the UDP protocol.

A live TV channel group is created with a UDP multicast source address.

The group is then shared across the access network with members (consumer devices) having the option to join the group to download the relevant data packets from their nearest accessible network node using the IGMP protocol. As multicast is a connectionless transmission method, the sender / source has no relation to the member downloading the data packets. The sender will not resend any data packets to a member. The main

benefit of multicast is that each source is only transmitted once which offers significant efficiencies for the

ISP. It is also a fast transmission protocol as there is no overhead to check if a member received the packet. The main drawback of multicast is the lack of acknowledgement that the member has received the data packets. This lack of acknowledgement and high throughput demand also renders it particularly unsuitable for wireless transmission, be that via mobile networks or in home Wi-Fi networks. Also, multicast is unsuitable for non linear features such as catch-up TV, nPVR, VOD etc. given its one-to-many transmission method. These features require a one-to-one transmission method such as unicast in order to support trick play which enables the user to pause, fast forward and rewind the stream which would be essential for non-linear features.

method, it is generally unicast which is more

(IP) Unicast is a connection-orientated, one-to-one transmission method built on the TCP/IP protocol. A live TV channel is assigned a URL and made available to consumers to initiate a stream request using HTTP / HTTPS. Once a stream request has been successfully established, the relevant fragments are placed on an identified server for the consumer's end device to download.

Each packet download is acknowledged by the end device, with the ability for the device to specifically request missing packets where necessary, hence the connection-orientated nature of unicast. The main benefit of unicast is the ability to guarantee that the receiver has downloaded the required packet. It also lends itself equally to both live and non-live use cases as it is the consumer's device that initiates the request. Additionally, unicast by virtue of its connection-orientated nature, is especially suited to streaming devices with a heavy dependency on wireless networks such as mobile data and WiFi to consume video content. This is further complemented by its ability to offer ABR streams to further enhance wireless delivery. The main drawback of unicast is its lack of efficiency when it comes to delivering live TV within the ISP's core and access networks to serve all consumers wishing to watch the same content. When considering how latency impacts each transmission impacted. Unicast delivery, unlike multicast, tends to be accompanied by ABR which adds the need for more latency on the video playout buffer belonging to the media player on the consumer's end device. Although multicast has an ABR implementation which has been adopted by a number CDN suppliers, it introduces its own share of video latency by virtue of the multicast-to-unicast conversion that must take place on the multicast-to-unicast agent running on the consumer's home gateway. Devices within the home request unicast live streams which the agent interprets, acquires from multicast sources, fragments and then formats into the required unicast stream.

Lastly, unicast depends mainly on the fragmentation of content before being encrypted and packaged. Multicast does not require the same fragmentation rather just packaging of the content for distribution. UDP Packets can be sent and downloaded out of sequence by members with only basic reference checksums to determine the correct packet order. As such, a media player supporting UDP multicast has to do far less work to access the packet's payload to get it ready for playout. Also it does not need to consider any bandwidth checks to switch profiles for ABR as that is not supported. Therefore, the time between

downloading UDP packets and playout is generally shorter than that of a unicast model.

Now that we have defined in detail the factors that can cause delay with IP video stream technology, we will turn to what Zattoo has done to deliver our low latency feature.

Our Goals

By 2026, it is expected that more Pay TV subscribers in Europe will use some form of IP connection as the source for their Pay TV services as opposed to those using a satellite or cable connection. IP-sourced Pay TV subscribers will rise by over 4 million to an estimated 47 million, whereas satellite and cable subscribers are expected to fall by 7 million to 18 million and 37 million subscribers respectively. If this trend continues, by 2030, Pay TV subscribers using an IP connection will outnumber Pay TV satellite and cable subscribers combined.

In anticipation of this shift, Zattoo began working over two years ago on a solution to reduce video latency. Our goals were to either match or better the video latency of our unicast based service compared to other IP based services, including those delivered using multicast. We set ourselves the target to get our video latency down to below 10 seconds for the majority of our supported devices.

Key to the delivery of this target were a number of conditions. Firstly, we would continue to use unicast

as our transmission mechanism. Secondly, we would make our solution technique independent, i.e. the same solution needed to be usable on HLS and MPEG-DASH and not require unique adaptations which favoured one over the other. Lastly, the solution had to easily scale to all Live TV channels on our platform and re-use our existing infrastructure.

In setting these targets, it was important that they both be achievable, yet push for the best possible outcome. Setting a target to match the video latency of our service to that of Satellite TV, although not impossible, would have been practically unachievable due not only on the conditions referenced above but also on the limitations that existed within technology itself, outside of our control. On the other hand, pushing for the best possible outcome, was to ensure that work undertaken challenged how we handled video processing and

delivery up to that point, where and how it could be improved and how it could be assessed. The later sections of this White Paper will reference a number of these challenges.

Our original goal of launching the low latency feature was to have the solution available for a select number of channels in time for the UEFA Euro 2020 Football Championships. Unfortunately, the COVID-19 pandemic forced the postponement of the tournament until the summer of 2021. We took the decision not to proceed with the launch of the feature in summer 2020, but instead to refocus our efforts to refine the solution. With the impact the pandemic was having on sports events in general, not to mention the impact on governments and people, it became apparent that anticipation building of the UEFA Euro 2020 Football Championships being held in the summer of 2021 instead would result in it being a truly massive event. Audience viewership is expected to reach 5 billion worldwide, a rise of 3 billion

vast majority of this increase can be attributed, not to how many people will watch the tournament, but how they will watch it. With varying levels of restrictions on movement and gatherings, more viewers will watch the tournament in some degree of isolation. What was previously an opportunity to gather with friends and family to cheer on your team, will likely be replaced with many more smaller gatherings and use of other forms of remote social contact to share experiences.

Given these unique circumstances, it further validated our decision to take the time to refine and fully prepare our solution. In May 2021, we released our low latency feature across all channels to all direct to consumer customers and B2B TV Platform tenants of Zattoo's platform.

since the tournament was last staged in 2016. The

Technical Deep Dive

Fragment Length Reduction

For many years, we have been using fragment lengths of 4 seconds across all streaming use cases, both live and non-live, to all of our customers of our Direct to Consumer service and end users of our TV Platform operator customers (B2B). When compared to the

fragment lengths of competing products and services, 4 seconds is on the lower end of the scale. We considered 4 seconds represented a fair medium between the overall size of the fragment and the consistent experience afforded the end user before a change in stream quality, if network conditions deteriorated. Although it is true that a far longer fragment size maintains the end user's quality experience for longer, the downside is that it takes far longer for the fragment to download. Selecting a lower fragment length requires the media player on the application to request fragments more often which adds additional load on the fragment dispatcher, but enables the media player to start streams faster and adapt to higher quality profiles at quicker increments.

In selecting a 4-second fragment length, we also took into consideration the access network through which the fragment may be delivered. For example, Zattoo's direct to consumer service in Germany, Switzerland and Austria is an OTT service. Customers access the service using their own ISP. Zattoo, for its part, peers with those major ISPs in order to deliver the service with minimum peerings and added network latency. For Zattoo's TV Platform operator customers, the service is primarily delivered to the operator customers' end users using a private peering between the Zattoo TV Platform and the operator's own managed network. Caching servers are also deployed where required within the operator's network to achieve further efficiencies. In a number of cases, network operators also offer OTT or "off-net" rights to enable their end users to consume the multiscreen part of their TV service, or part thereof, using another ISP's network. Similarly, providers of online content services, of which network operators may also be classified as, were required to extend such "off-net" rights to allow end users of the provider to consume paid content in any EU member state when the EU introduced cross-border portability regulations in 2018. As such, it was imperative that the fragment length could meet the demands such a variety of access networks could introduce, whilst also delivering a smooth and consistent user experience.

Since the introduction of the 4-second fragment length, we have invested considerably in our streaming infrastructure and content delivery network. We now offer, and strongly recommend all operators, to deploy the aforementioned Zattoo supplied caching servers within the operator's own managed network. We have also invested considerably in our interconnect infrastructure which has meant that we can establish a wider network of private peers or transit links towards other network operators, ISPs or Internet Exchanges within the relevant regions quickly and efficiently. In essence, the overall performance and reliability of our content delivery networks and infrastructure has increased significantly.

Additionally, since the introduction of the 4-second fragment length, we have added substantially to the quality profiles of our streams. In one specific case, we offer two independent quality profiles at the same resolution with merely a reduction in frame rate which enables the lower frame rate profile to be offered at a lower bitrate. Therefore, where the end user does encounter a degradation in their network connection speed, the likelihood is that the media player on their application will first adapt to a quality profile with the same resolution and therefore be far less noticeable, if at all, to the user that a change in quality has occurred.

Spurred on by our low latency goals, the time was right to reduce our fragment length to 1.6 seconds for all streams. In arriving at a 1.6-second fragment length, in addition to taking all the above improvements into account, we also assessed what the best fragment length would be for our encoding infrastructure. As our encoders are wholly Zattoo in-house developed and maintained, this required a significant amount of testing and re-testing of the encoding infrastructure for performance implications.

Another benefit of the improvements which enabled the reduction of the fragment length was the opportunity to reduce the buffer we use within the encoding process itself. Previously, we would buffer up to four fragments of content within the encoder to enable apt time for the transcoder to perform its task of producing the lower quality profiles. We now

no longer needed as much encoder buffer size as the fragment length is lower, and can also reduce the number of fragments buffered to three, which improves the overall performance and efficiency of the encoder.

The reduction of fragment lengths also brought a significant benefit to stream start-up and switching times. As the fragment lengths are smaller, it takes far less time for the media player on the application to download the fragment and send it to its video playout buffer as expected, which resulted in an overall reduction in stream start up times.

In early 2021, we began migrating all of our encoders and streaming infrastructure to support the new fragment sizes. The migration was completed in the middle of Q2 2021.

Smart Buffering

At Zattoo, we develop our end user steaming applications compatible with a large number of mobile, big screen and Set Top Box devices. The majority of our applications are developed natively using the SDK's and toolsets stipulated by the device manufacturers, i.e., iOS, tvOS, Android Mobile, Android TV, etc.. We also develop a web application that, in addition to offering broad capability for all the market-leading browsers, is also portable to many other device types such as Smart TV's and game consoles. We maintain this suite of applications both for our direct to consumer business and as a white label for our TV Platform operators. For each application type, we select a media player to perform the content playout tasks. In the case of iOS, tvOS and certain Smart TV devices, we use the native media players supported by those devices. For Android and FireTV, we use Google's ExoPlayer. For the remaining applications, we generally use a commercial media player.

Earlier in the section on understanding video latency, we described the role the applications' media player performs when it relates to content streaming. To quickly recap, the media player's role is to initiate the downloading of IP packets containing the required video content fragments and store them in the video buffer for playout. The video buffer is a portion of a device's memory that is used to store video or graphics information as it moves from the renderer (video chip) to the display output. The buffer is created and controlled by the media player. The application uses the media player's integration interfaces to interact with it and, in some cases, change the configuration parameters of the media player to operate as defaults.

As already outlined in this section, we use a number of different media players. As you would expect, the media player's configuration and interface capabilities differ somewhat from each other. Therefore, up to now, we have generally opted to use the most common / default / natively-supported parameters of each media player. This also includes how the media player manages the buffer before rendering to the display output. It was clear that in order to deliver on the low latency goals, we would need to find a way to adapt the buffer configurations of the media players available across all applications.

The media players used by our applications do not all manage their buffer in the same manner. Some apportion the buffer based on the number of fragments needed to be filled before playout, whereas others apportion it to a number of milliseconds

needed to be filled. So in proposing changes to the buffer configuration, we would need to make such

adaptations on a per media player level, and to test for consistency of experience across all applications, even those using other media players. Additionally, and perhaps most importantly, we could not lose sight of the importance of why the video buffer is there in the first place to enable a smooth and consistent experience for the user, even where stream quality changes are being invoked. Underresourcing the video buffer would only lead to a poorer end user experience, especially where challenging network conditions are encountered.

Therefore, it was apparent that we needed two elements to feed into the decision process when considering whether to increase or decrease the size of the buffer at the start of each playback. The first would be a simple report of any buffering events that occurred during the previous stream playback, i.e., if the buffer was too small, we should expect to see at least one buffering event and therefore should increase the buffer size for the next playback. The second element would be some form of learning algorithm which would use three weighted inputs to effectively score the user's past streaming experience. The weighting of the inputs would also need to be configurable to meet the type and profile of the application, i.e. devices that offer a LAN connection would likely have a more stable internet connection than those with a WiFi connection only, for example. The algorithm inputs would be:

- **Last Playback Buffer:** The size of the buffer used during the last stream playback on the application.
- **User Session Buffer:** The average size of the buffer since the start of the application session.
- **Global Device Buffer:** The average size of all buffer values used by the user on this device.

Underpinning the whole scoring mechanism would be a principle in which the application is proactively attempting to reduce the buffer size in order to achieve a low latency experience during each new playback event. At the same time, the application is attempting to reward the scenario in which a suitably performant, reliable network connection has been proven, and therefore the opportunity is available to decrease the size of the video buffer.

We began testing the smart buffering configurations in a stepwise fashion and determined early on that the approach would definitely meet our goals of a sub 10-second video latency on the majority of media players. Unfortunately, we encountered significant challenges to achieve a sub 10 seconds on iOS and tvOS devices, instead managing to get to a 15-second latency and leaving room for further development and refinement. For those other media players, the ability to adapt the weightings on a per application basis was extremely helpful as it gave additional flexibility to adapt to the media players requirements.

Whilst the reduction of the fragment length feature could be comprehensively tested within our platform and content distribution network, it was clear that the learning nature of smart buffering would require a large part of the testing to be performed entirely in the field. This began at the start of 2021. By Q2 2021, the in-field tests were replicating the positive results from the internal testing and the feature was approved for deployment in the middle of that quarter.

For further detail on both of these features, please visit our tech blog on Medium.

Reaction and Future

The reaction to the introduction of our low latency feature has been very positive. The majority of our Direct to Consumer customers are using versions of our application in which Smart Buffering is available. Anonymised statistics shows the average latency across all applications either meets or exceeds the expected latency goal of 10 seconds for applicable applications, with iOS and tvOS devices achieving sub 15 seconds latency. TV Platform operator customers have a similar usage share with identical statistical data in terms of experienced latency. Reaction from the industry has been equally positive and we are receiving considerable interest from prospective operators interested in joining the Zattoo TV Platform.

In terms of future developments, we intend to further improve the latency on iOS and tvOS devices to bring it in line with the rest of our applications. We also intend to further refine the weightings as we learn more and more about the experiences of end users.

We believe that we have taken the first big step in reducing our video latency and what will follow will be subsequent smaller steps as we adapt and refine the feature helped on by technological advancements. We expect to continue to challenge ourselves with measurable and achievable goals in this regard and we are confident we have the means and dedication to achieve them.

About Zattoo

Zattoo strives to satisfy existing customers and attract new ones by offering state of the art TV features and functions with a superior user experience across many devices and device platforms. Our product portfolio ranges from back end services such as ingest, encoding, and transcoding to hostedand managed end to end solutions for first screen IPTV & OTT and second screen mobile/web TV devices such as Android TV for Operator set-top boxes, as well as Apple TV as both a set-top box and retail device, Amazon Fire TV, Smart TVs and mobile devices (iOS / Android / Windows 10). Zattoo gained recognition for its achievements in 2020 with the awarding of the Technology and Engineering Emmy® Award from the National Academy of Television Arts and Sciences.

For more information visit **thetvplatform.zattoo.com** follow us on **LinkedIn** and **Twitter**

EMMY®AWARD 2020

Zattoo gained recognition for its achievements in 2020 with the awarding of the Technology and Engineering Emmy® Award from the National Academy of Television Arts and Sciences.

References

Web

https://medium.com/@stefan-kaiser/the-definitive-guide-for-picking-a-fragment-length-617f75b9ccf3
https://medium.com/@stefan-kaiser/how-to-go-low-latency-without-special-tricks-37c69db027bb
https://medium.com/@stefan-kaiser/smart-buffering-and-the-two-types-of-player-configurations-9cb052d34828
https://www.imglicensing.com/client-portfolio/uefa-euro-2020/
https://en.wikipedia.org/wiki/Satellite_Internet_access#Signal_latency
https://wondernetwork.com/pings

Documents

Western Europe Pay TV Market Forecasts to 2026