iab. TECH LAB

Podcast Measurement Technical Guidelines
Version 2.1

Released March 2021

The IAB Podcast Measurement Technical Guidelines document was developed by the IAB Tech Lab Podcast Technical Working Group, in partnership with the IAB Audio Committee. The following IAB Tech Lab member companies contributed to the creation of this document:

- Acast Stories USA
- AdGear Technologies, Inc.
- AdLarge Media
- Adswizz Inc
- Art19
- Audible.com
- BlogTalkRadio
- CBS Local
- Condé Nast
- Cox Media Group
- Cyber Communications Inc.
- Digital Advertising Consortium Inc.
- DoubleVerify
- ESPN.com
- Libsyn
- Midroll Media

- Minnesota Public Radio
- NPR
- New York Public Radio
- Nielsen
- Pacific Content
- Pandora
- PodcastOne
- Podtrac
- RawVoice/Blubrry
- RhythmOne
- Sizmek
- Slate
- Triton Digital
- Westwood One
- WideOrbit
- Wondery

https://iabtechlab.com/working-groups/podcast-technical-working-group/

The IAB Tech Lab lead on this initiative was Mike Midden.

Please contact audio@iabtechlab.com if you have any questions or comments about this document.

This document (and future updates) can be found on the IAB Tech Lab website at:

https://iabtechlab.com/standards/podcast-measurement-guidelines/

Table of Contents

1. Executive Summary

Podcast audiences represent a growing segment of effective marketable media. Podcast ad revenues are expected to reach over $1 Billion in 2020, a 14% increase from 2019, according to the IAB Podcast Advertising Revenue Study conducted by PwC US. Podcasts offer advertisers a hyper-focused audience that chooses to listen to that content. Podcast listeners have been shown to be the most loyal and engaged audience of any digital medium. However, measurement of content and ad consumption in podcasting has not been consistent so far, which sometimes limits participation of brand advertisers. This document provides an introduction to tracking ad delivery in a podcast and attempts to provide clarity in the marketplace by describing best practices for measuring downloads, audience size, and ad delivery.

Podcasts are downloaded to a device for later listening or through progressive download listening. In most cases the podcast file and any ads included with it are downloaded to a device that doesn't, or can't, send data about the consumption of the podcast and ads. This lack of data beyond ad delivery limits real-time measurement. In contrast, other media are consumed by reading an article and interacting with a site, playing a game, or streaming a video, all of which can be measured in real time. Even audio stations that offer music or news are streamed and measured in real time in today's media marketplace.

Podcast listeners have the ability to download files to consume whenever and wherever they want and are not required to have an active internet connection to play back an episode. The medium, the distribution, and the platforms used to collect and listen are built around the habit of downloading the file. Tracking content in this time-shifted medium involves filtering server logs to produce meaningful data for measurement. Since podcast technical teams analyze server logs differently, results vary across the industry.

The challenge for podcast producers and distributors is to offer buyers a set of metrics that is consistently defined and measured equally across the podcast medium. The definitions in this document aim to reduce measurement discrepancies and present a set of recommended metrics and guidelines based on industry best practices. With a consistent set of podcast advertising metrics, buyers and sellers can engage in a conversation about campaign strategy with confidence.

Audience

While all professionals in the podcast supply chain can benefit by being familiar with this document, metric definitions are primarily intended for podcast producers and distributors. Specifically, account managers should be familiar with and use metrics as defined in this document when negotiating ad packages with buyers. Additionally, podcast ad operations teams should use the metric definitions in this document to design or adjust the ad measurement technology they use to analyze server logs for podcast ad measurement.

Buyers should also reference this document to better understand how ads in podcast content are counted. This document offers a set of metrics that establish a mutual understanding in podcast advertising negotiations.

About this Version

The first version of this document was released in September 2016, and additional significant updates were made with version 2.0 released in December 2017.

Updates in this version (2.1) includes language edits for more clarity, a new section with guidance on user agent structure, recommendations for IPv6 IP addresses, filtering guidance for Apple watchOS user agents, and a number of additional podcast player recommendations.

2. Overview

Podcast content is an on-demand media format that listeners either download to listen to later or consume online. Unlike the streaming format more common in video, podcasts continue to be downloaded because of the convenience offered by existing platform and application functionality.

Despite the use of the word "streaming" in podcasting, "streamed" podcast files are progressively downloaded via the standard HTTP protocol. True streaming—typically reserved for live events—requires a specialized server and uses an entirely different protocol.

While "streaming" a podcast and true streaming formats appear exactly the same to end users, delivery of a streamed podcast is logged the same way as a downloaded file in the server logs. This important distinction impacts the ability to measure content and ad delivery in real-time without access to client-side analytics. Podcast publishers must work around this limitation and track metrics using server log data.

Because log-based measurement counts only the file downloads and not the actual listening, in the long run it might not represent the best possible measurement approach. New player-based measurement approaches recently announced by Apple and others show the potential for podcast measurement that's in the client app and can track actual plays, pauses, and other listener behaviors. But it will take some time for any of these client-side approaches to be adopted by all the podcast players that exist, and will depend on the extent of data made available. So for the foreseeable future, log-based measurement is the only way to achieve comprehensive measurement of all podcast usage.

Media delivery via true streaming falls outside of the definition of a "podcast" and is therefore excluded from this document.

Podcast Player Market Share and Tracking Limitations

The ability to track podcast content and ad playback largely depends on the player requesting the file. The native players that operate on iOS systems, namely the Apple Podcasts App and iTunes currently do not offer technology to confirm that a podcast file was played. While this hopefully will change with Apple's announcement of client-side metrics, at the time of publishing this document, this lack of client-side response prevents podcast distributors from measuring ad plays at the level expected in other digital media. Once Apple and other platforms start supporting client-side metrics, the Podcast Technical Working Group will work on an update to these guidelines to include those.

In order to provide insight into the limits on tracking podcast content, the IAB Podcast Measurement Working Group was asked to provide reports on the market share for platforms that request podcast files. Podtrac, NPR, Blubrry/RawVoice, Libsyn, and Triton submitted reports used to produce the following table, which aggregates the resulting United States market share download percentages for 2020.

Aggregate Report on Podcast Player: Market Share of Downloads (2020)

Platform requesting podcast file	Average market share %
Browsers	5%
Spotify	10%
Everything else	26%
iOS - Apple Podcast App / iTunes / AppleCoreMedia	59%

Reports for non-iOS (Apple) market share varied from one report to another, but the results reported for the iOS devices were consistent across all reports, with a mean of about 59%.

Reports for Spotify were generally consistent as well, coming in at an average of about 10% of the market share. A major caveat for Spotify though is that the total market share may be under-represented, as the Spotify Passthrough feature which allows for the measurement of Spotify hosted podcasts has not been implemented by many small-medium sized podcast publishers resulting in partial visibility to Spotify's footprint.

For about half of the podcast ads served to browsers (5%), some servers may be able to distinguish ad delivery from a probable ad "play." Using browser plugins or other technology, a specialized tag used to request the ad file can indicate that the player accessed the ad. While this technique offers valuable tracking data, half of the 5% means that currently only about 2.5% of podcast ads can be tracked this way.

Another key insight for this report (not represented in the table) is that only a very small portion of the market share enables client-side tracking as it exists in other forms of digital advertising. Host-branded players (players owned by the podcast producers) are a rarity in the industry today. In these instances, the producer controls the player and the content, allowing them to request ads and trigger tracking beacons based on ad play. Most podcast distributors see almost zero activity in this market. Even of the few host-branded players that exist today, many may not be equipped to find tracking beacons and use them. Future focus of the Podcast Technical Working group will explore expanding ways to access client-side data and the viability of applying measurement standards such as Open Measurement.

The data provided for this report shows that more than half of the market share for podcast players belongs to iOS devices, which prevents any client-side tracking or even the ability to count a "play." Podcast distributors must turn to server log analysis and report on ad delivery. Some distributors count an ad once it's been served. This count offers a valuable metric, but is out of scope for this document because an ad served doesn't indicate whether the ad file was downloaded.

Despite the limitations, podcast audiences are growing and offer valuable exposure for marketers. In order to offer this value to buyers, metrics must be consistently defined across the industry. IAB collaborated with members in the podcasting community to establish metric definitions that can be used consistently in the podcast marketplace.

Establishing consensus and clarity for podcast reporting metrics improves communication and establishes trust and accountability with buyers.

Scope

This document defines content, ad, and audience metrics in the context of downloaded podcasts whether saved for later listening or listened to while being downloaded. In this context, both formats are typically pre-recorded and available on demand whenever the listener is ready to access the files.

Podcasts that use streaming technology to deliver the ad offer the ability to track activity in real-time or near real-time and one metric used to measure "client-confirmed ad delivery" is covered in this document. However, the percentage of market share for applications that support true streaming is currently too small to account for any meaningful campaign measurement. Additional measurement guidance for true streaming audio is covered in the MRC Audio Measurement Guidelines currently in development. Those guidelines also cover client-initiated podcast measurement, which is not the focus of this document.

Ad measurement in podcasting presents the industry with many challenges. For the sake of establishing common ground in tracking podcast ads, the definitions presented in this document address counting ad delivery. This count comes from analyzing server log files to determine what was actually delivered.

3. The Podcast Medium – Content Delivery

Podcast listeners acquire podcast files in one of two ways: either by downloading the file for later listening (downloaded), or by listening while the file is downloaded (online listening). To a lesser degree, some podcasts may also be played while a persistent connection to the server is maintained (streamed), but the market share for applications that support this format is insignificant for campaign measurement and excluded from discussion here.

Delivery methods for downloaded files, whether listened to later or during download, offer valuable inventory to advertisers, but content and ad delivery are handled differently in both environments. An overview of each format is explained in the following paragraphs. Despite different tracking capabilities in each environment, a few baseline metrics should be able to offer similar reports for both podcast types.

Downloaded Podcasts

Podcast downloading allows the audience to download full episodes of content that can be played at a later date and time. Listeners may subscribe to select programs, and platforms like the Apple Podcast App continue to support full downloads to a personal library for listening offline at any time in the future. The convenience of this system makes downloaded podcasts a continued preference among listeners.

Progressively Downloaded Podcasts

These podcasts appear to be streamed, but the file is actually being downloaded while the listener is listening to the file. The downloaded file is stored in a temporary location rather than to a library as with a downloaded podcast. Since progressively downloaded files are typically downloaded the same way as the files stored for later listening, delivery for these two formats are recorded the same way in the server logs. The only difference between the two is whether the listener is actively playing the file as it is downloaded or being saved for later listening - which can only be discerned by the player.

Raw Server Logs

In a downloaded file, segments of the file are collected on the listener's device, or progressively downloaded. These progressively downloaded files result in a server log with several requests to the server, which must then be analyzed and filtered from other server requests in order to represent how many files were downloaded and to what audiences. When podcast publishers use a consistent process, metrics can be reported and trusted with a higher level of confidence.

4. The Podcast Medium – Ad Delivery

Podcast ads can be delivered and tracked in a variety of different ways, but in general two different methods are used with variations on each.

Integrated Ads

Historically, podcasting ad campaigns often involve ads that are read by the podcast host or a familiar voice. A static ad or jingle may be also included as part of the file. These ads are part of the content and included, or "baked-in," with the file that is downloaded. Targeting is limited because everyone who downloads the file gets the same ads.

Dynamically Inserted Ads

In recent years, ad technology has allowed for ads to be targeted and dynamically inserted at the time of file request. The ad server determines the best ad to serve to the listener at the time of request. In a podcast consumed online, ads may be inserted into a file that is being progressively downloaded at designated ad breaks. Some publishers may count this dynamic ad serve as an "impression" without confirming ad delivery. The metrics in this document focus on confirming that the ad was delivered. Server logs can confirm that the entire ad file was downloaded, but the process for counting a served ad can only determine that ad file was sent.

5. Podcast Measurement

In digital display advertising, ad tracking is performed using beacons that are triggered in the web browser, or client, which verifies that the ad was presented and at least had an opportunity to be viewed. In podcasting, client-side tracking is usually only possible when the client player passes tracking data back to podcast producer or distributor. In this set-up, the player is programmed to notify the server when an ad has been played. While this set-up offers the most accurate ad delivery counts, it currently represents a very small percentage of the podcast industry–less than 3% according to member reports on market share in the industry (see table 1).

Measurement with Server Logs

In order to produce accurate counts for podcast downloads and ads, technical staff in publisher and podcast ad operations must analyze server logs. These server logs may include file requests for a combination of downloaded podcast files, dynamically inserted ads, and any content requested by the web page or application hosting the player. A number of factors are used to analyze log files.

HTTP GET requests may be processed that contain the following data.
- **IP Address** - The IP address is one of the factors that may be used to determine if the request is unique or a duplicate. (Exceptions are shared locations such as corporate offices, dorms etc., that have a large number of people sharing the external IP Address) It may also be used to determine geographical information of the media consumer. This applies to both IPv4 and IPv6 IP formats.

- **Time Stamp** - The date and time may be used to determine if the request should be counted.
- **HTTP Status Code** - The appropriate HTTP status code is examined to determine if the request should be counted.
- **Bytes Served** - The value may be used to determine if the media was completely downloaded or if not, how much was downloaded. (Note: This information is only available from native server log files.)
- **Referer** - The origin of the download may be used to determine if the request should be counted. e.g. media that is auto played upon loading a web page may be removed or reported.
- **User Agent** - The identifier of the application or service consuming the media may be analyzed to determine if the request is unique.
- **Byte Range** - The range of bytes requested in a given request may be used to determine what portion of the media is requested.

When analyzed across multiple requests, the information may offer statistics that represent podcast downloads, audience and ad delivery. Since media technology is always changing, no specific combination of factors or techniques will offer the most accurate count indefinitely. However, meeting some minimum requirements and following some best practices will help produce more consistent results. The next section will go over some best practices for generating server side metrics.

The document then defines a few metrics for podcast content measurement, audience measurement and ad measurement. Podcast producers and distributors may include additional metrics beyond the ones defined here, but such additional metrics should be labeled separately from the core list of metrics described in this document.

Recommended Process for Measurement

This section lists best practices based on the experiences of the members of the Podcast Technical Working Group. While we have made the effort to be specific, publishers and distributors will have to look at the various options available and select the best for their particular circumstances. Metrics providers should support the process below or a process with a similar or more stringent level of analysis sophistication, disclose the options selected, disclose where they diverge from the recommendations, and provide the rationale/circumstances that drove those decisions.

To claim compliance with these guidelines, an organization must go through the IAB certification process and get listed on the IAB Tech Lab website:

https://iabtechlab.com/compliance-programs/compliant-companies/#podcast

We recommend a 5-step process to generating metrics using server side log analysis.

1. Apply filtering logic
2. Apply file threshold logic
3. Identify and aggregate uniques
4. Generate metrics
5. Audit the process (feedback loop)

These steps are described in the following pages. The recommendations assume a calendar day 24-hour window, in the time zone as chosen by the org for calculating the metrics. No window is perfect, but shorter windows open up a risk of double counting requests and so should be done with care. Conversely, longer windows risk undercounting delivery via recycled mobile IPs and true multiple listens. Companies are allowed to use more sophisticated mechanisms (like a rolling 24-hour window), but we are not mandating that because of the level of complexity that could introduce, with limited benefits.

Step 1. Filtering

All requests that should not be counted for any reason should be filtered out up front. The criteria we have identified for filtering are listed below.

1. Eliminate Pre-Load Requests
Pre-loading of podcasts directly results in podcast downloads being counted when they should not. There are two possible solutions to handle this.

2. Policy put in place to not allow pre-loading in players and on websites (e.g. preload=none for HTML5)
3. Use a download threshold based on ID3 header payload plus 1 minute of recording time to determine if request was for a play/download or for pre-loading (see Step 2 **"Apply file threshold levels"** below)

2. Eliminate Potential Bots and Bogus Requests

There are a number of scenarios where the raw requests include requests that should not be counted because they likely come from bots or from products that behave in ways that make them look like real downloads. We recommend that metrics providers look for the following to filter out.

1. IP addresses that cannot originate to actual users (for e.g., known servers)
2. IP addresses that account for a large number of downloads should be examined for potential fraud. (But also look at the safe IP addresses note below.)
3. IP addresses that are on a service like AWS.
4. Erroneous referrer data
5. Bogus user agents, e.g. Firefox 3.06
6. User Agents that identify to be from sources that are not actual users (e.g. bots that selfidentify as being bots)
7. Similarly, referer data that implies that the sources are not actual users.
8. Apple clients – Official Apple iOS Podcast app performs a 2 byte range (Range: 0-1) request that should always be excluded from processing. This request is made by Apple to check that the media file can be downloaded using byte range requests and is immediately followed by 1 or more additional range requests. Best practice is to ignore 0-1 byte range requests.

> **Note:** Known "safe" IP Addresses (dorms, corporations etc.) should be maintained in a inclusion-list and be allowed through. These likely need to be re-validated frequently (say every 30/90 days) since IP addresses may not be static.

> **Note:** The members of the Podcast Technical Working Group also suggested that we should start a project to share an inclusion-list of "safe" IP addresses that represent dorms/corporations and other known NAT situations. Similarly, we likely need an exclusion-list of known bad IP ranges. We will look into these after publication of this document. These lists should not be public and only accessible to members so as to prevent the bad players from adapting quickly based on that information.
>
> We are not developing these lists right now, and leaving it to the measurement platforms to manage them themselves. We might pursue these efforts in the future if there is sufficient interest.

3. Handling HTTP Requests

This section covers the correct handling of the logs based on the various types of HTTP requests.

1. HEAD requests - these should not be counted because this is typically used to check for changes because no data is transferred in a HEAD request.
2. GET requests –
    a. 200 (ok request) should be counted
    b. 206 (partial request) A partial request should only be counted if the download covers the 1 minute rule, and de-duplication based on IP Address/UA is being done to cover cases where the user might be skipping ahead. Determining whether the requests cover the 1-minute requirement might require reassembling of the requests.
    c. 304 (not modified request) -> signal that user has existing file and wants to see if it changed.
3. There may also be platform specific quirks to watch for. For example, Akamai uses a HTTP code of 000 for 206 requests that ended prematurely.

Step 2. Apply File Threshold Levels

Downloads below a certain size are unlikely to result in human consumption because too little of the file was received to listen to any content. The following rules help eliminate the downloads that are too small to be counted.

1. To count as a valid download, the ID3 tag plus ***enough of the podcast content to play for 1 minute*** should have been downloaded.
2. ID3 size recommendation – since the ID3 file size can vary quite significantly, each publisher should measure the ID3 tag file size for each podcast. To be more efficient in cases where the ID3 size doesn't change, the publisher could set a size for the show/program and whenever the artwork changes, re-calculate the size.
3. Content size recommendation – the size of the download for 1 minute of content will vary based on the bitrate used and the amount of bytes ID3 headers consume. So our recommendation is for the publisher to calculate this size for each show.

This step requires a continuous monitoring of the podcasts as each episode gets served.

Alternatively, if the podcast is too small or if it isn't possible to compute the file and ID3 sizes regularly, *complete file downloads* (100% of the file, including the ID3 tag) should be used.

> **Note:** One minute was chosen as a conservative minimum size since other mediums use similar or smaller thresholds.

> **Note:** Byte range request data is not available: If the logs do not have byte range request data, more advanced algorithms that factor in a correction for partially downloaded content may be used. Such a system must disclose how their system overcomes not having the byte range data.

Step 3. Identify and aggregate uniques

Once filtering is completed, requests should be aggregated to identify uniques in the following two scenarios.

a. **Identifying Uniques (for Downloads & Users):**
Identifying unique requests is important in counting downloads for an episode and in counting audience size. The following method is recommended, and the details of the filtering methods should be kept transparent.

Filtering using IP address + User Agent

- A combination of IP Address and UA is used to identify unique users and downloads. For example, if the same file is downloaded 10 times by 6 user agents behind one IP address, that would count as 6 users and 6 downloads.
- This method requires some technique to constrain counts for an exclusion-list maintained to block IP addresses that excessively download/play at a rate that is not feasible.
- In order to better support known high density IP Addresses (dorms, corporations, etc.), an inclusion-list of IP Addresses may be maintained. For these IP Addresses, different filtering rules may be needed to account for a concentration of similar devices.

b. Play-Pause-Play Scenarios

If a unique download is divided into multiple file requests, for example if a user plays the first half of an episode using a website audio player, clicks pause, and then resumes a half-hour later, then that should still be counted as one unique download.

Care should be taken to not count these as multiple downloads/users.

Step 4. Generate Metrics

Once the requests have gone through the filtering process above and uniques have been identified, it is time to generate the metrics defined below, as well as any additional / custom metrics supported. We will not prescribe how the metrics should be formatted / delivered or recommend any particular analytics technology over the other.

Step 5. Audit the Process

The goal of this section is to allow for adjustments to the metrics generation. Metrics platforms need to watch for behavior that indicates that the quality of the metrics is diminished, and investigate the source of potential errors / fraud.

We recommend that the entire process should be self-audited on an ongoing basis. Red flags should be identified and metrics adjusted based on a deeper investigation of the red flags. In addition, future runs of metrics generation should factor in any learnings from each run. For example, if certain IP addresses are identified as being potential bad actors, those should be eliminated in the metrics of the current run and should also be placed in an exclusion-list for future rounds.

A good method to audit metrics would be to compare the bandwidth usage to the metrics. The deltas should be linear.

Podcast Content Metric Definitions

Since podcast ads are so closely integrated with podcast content, metrics that measure content are vital to ad measurement in podcasting.

Show producers, executives, marketing, and digital product teams are interested in the following questions:

- Audience: How many people are downloading my network/show/episode?
- Downloads: How many times is my network/show/episode downloaded and potentially listened to, at least in part?

Podcast Delivery Metric Definitions

The following metrics are used to describe content downloads. Server log analysis for content delivery should filter data to produce metrics as defined below.

1. **Download:** a unique file request that was downloaded. This includes complete file downloads as well as partial downloads in accordance with the rules described earlier.

Once we introduce the "Confirmed" plays from player side listener tracking, we likely can reuse the same set of metrics and add the term "Confirmed" to them, showing both total and confirmed side by side.

Publishers can opt to also provide non-filtered (non-unique) file request information to offer a gross consumption metric.

Podcast Audience Metric Definitions

Podcast listeners often download more than one episode and often from more than one podcast. A measure of how many people downloaded episodes can be used to describe the reach of the podcast or group of podcasts.

Website metrics of reach usually depend on the use of cookies or JavaScript to identify individual devices or users. Downloading an MP3 file does not usually trigger the reading of cookies or the running of JavaScript, so alternative methods must be used. HTTP requests

include the IP address of the recipient of the file and usually include a user agent that, while not unique to the user, provides some ability to distinguish multiple users behind one IP address.

2. **Listener**: data that represents a single user who downloads content (for immediate or delayed consumption). Listeners may be represented by a combination of IP address and User Agent as described earlier. The listeners must be specified within a stated time frame (day, week, month, etc.).

> **Note 1:** if the metrics provider is able to identify listeners by cookies, IFA or other similar mechanism that requires more advanced client support, they are encouraged to do so. The key requirement is to be transparent about the methodology used. It is important to note that as discussed above, currently only a very small subset of podcasts consumed are covered by these mechanisms.

> **Note 2:** It is important to understand the limitations of the listener metrics in podcasting due to the nature of the usage. Most users are on mobile devices, which means that the IP addresses change frequently (potentially resulting in double counting users) and also could be recycled (potentially resulting in undercounting users).

Metrics for IPv6 Addresses

The members of the Podcast Technical Working Group recognize that measuring unique devices assigned with IPv6 addresses poses certain challenges due to the fact that IPv6 addresses are not static, with multiple new addresses being cycled on a singular device in a given time period. Without special consideration this characteristic of IPv6 addresses could result in inflated metrics, and could also impact frequency capping of ad campaigns.

The Podcast Technical Working Group has researched methods to address potential discrepancies attributed to IPv6 measurements resulting in the following recommended treatment for IPv6 based measurements.

IPv6 addresses should be truncated to its first 64 bits before calculating the Listener and Download metrics, which is a combination of IP+UA.

Listeners or **Downloads** = Count of Unique (IP* + UA)

*Either the full IPv4 or a partial IPv6 (prefix 64 bits).

Note that the IPv4 or partial IPv6 can be hashed for privacy reasons without affecting the above formulas. And though we can't stipulate that a truncation of IPv6 addresses to 64 bits makes the IUP+UA combination a bullet proof method to measure unique Listeners & Downloads, we have determined this methodology to provide results that are comparable to what is seen with non-truncated IPv4 calculations. Ultimately, as mentioned elsewhere in this document, the key requirement is to be transparent about the methodology used.

Podcast Ad Metric Definitions

The following metrics represent the first step toward improved ad measurement in podcast advertising. These metrics are derived using the content metrics defined above. As these metrics become adopted in the industry, additional steps can be made toward an improved podcasting ecosystem.

3. **Ad Delivered:** an ad that was delivered as determined by server logs that show either all bytes of the ad file were sent or the bytes representing the portion of the podcast file containing the ad file was downloaded.

   For example, if an ad was included within the first 25% of a podcast and at least 25% of the podcast file was downloaded, then the ad can be counted as delivered.

   When ads are dynamically inserted into the podcast file or within an ad break within the podcast, 100% of the ad content (all bytes) must be downloaded before it may be counted as delivered.

4. **Client-Confirmed Ad Play:** counts an ad that was able to prompt a tracking beacon from the client when the file was played. Whenever possible, metric should include information about how much of the ad was played using the markers: ad start, first quartile (25%), midpoint (50%), third quartile (75%), and complete (100%).

While the client-confirmed ad play metric represents the most accurate count for ad plays in a podcast, it requires client-side tracking. As discussed earlier, the platforms used to download, store, and play podcast files lack or prevent the technology needed for client-side counting. Aggregate reports on player market share among podcast publishers estimate that currently less than 3% of players are capable of providing client-side tracking data (see table 1). The IAB Tech Lab will continue working with the player platforms to get more access to client side beacons.

Higher Level Metrics

All the metrics in this document so far were written with a focus on episode level analysis. The Content and Ad metrics described above should also be made available at 3 levels – publisher / show / episode.

Client-side tracking or access to some sort of listener ID – for example using cookies or using an Identifier For Advertising (IFA) – would be the ideal mechanism to track audiences over time, but this can be approximated using analytical methods as described below. If the publishers / distributors have the ability to identify the user, they should indicate the mechanism used and provide the metrics (downloads, listeners and any other additional metrics supported) at the show and publisher level in order to provide a better picture of that podcast's reach.

Lacking a listener ID, there are a few options to build these metrics:

1. Sum the metrics across podcast episodes.
   - Acceptable if the goal is to just count the total number of unique downloads, but does not provide a view of the listener base due to audience overlap.
2. Use the IP Address & UA to identify and track users
   - Provides a better view of the user base, with certain limitations:
      i. The IP Address for a user could change – especially in the case of a mobile user – at which point there is no way to correlate the 2 IP addresses
      ii. Using UA helps differentiate multiple users from an IP address, but breaks down when multiple users from an IP have the same kind of device/UA (as is somewhat common in corporate and educational settings)
   - However, the above two negatives likely counteract each other over time.

6. Summary

Podcast audiences represent a growing segment of marketable media, and are considered to be some of the most engaged. However, the medium is asynchronous and in most cases severed from data collection once delivered, which presents advertisers with measurement challenges. In addition, measurement practices have been fragmented and ad-hoc so far. This document, with the **four** metrics defined above, offers the first step in an improved environment where buyers and sellers can start to use the same language with clearly defined meaning. As communication improves, producers will be able to scale their operations and invest in technology that brings tracking closer to the standards available in other media options.

7. Appendix: Podcast Player Recommendations

As we developed the podcast metrics guidelines, we recognized that it might be valuable to identify some recommendations that publishers factor in when they build their player / listener experiences.

1. Do not implement Auto-play. This will result in a bad user experience for the user with audio they were not expecting to hear.

2. Do not Pre-load - unless the intent was clearly to play the podcast.

3. Use ID3v2 tags, so that the headers are located at the start of the podcast (not at the end). This allows players to use the ID3 data ahead of streaming time without downloading the full podcast file.

4. ID3 tag sizes - recommend that the ID3 size be limited to 300kb with 800x800 px max for the art.

5. For a full download, ask for the entire file in one go. For a progressive download, ask for the file in slices (byte range). This way a full download can be distinguished from a progressive download.

6. Do not modify the enclosure URL when requesting media, don't add extra parameters.

7. Do not cache podcast episodes on your servers. Always download the latest episode from the enclosure URL for every app user wanting to listen.

8. Use the GUID -- as opposed to episode URL, title, publication date, etc. -- to identify new episodes in the RSS feed that should be automatically downloaded to a user's device. The GUID is designed to be persistent through changes to hosting environment, titles, etc.

9. Employ an "automatic download unsubscribe" behavior (e.g. - stop auto downloads after 5 episodes of non-listens).

10. Do not automatically download all episodes (e.g. back catalog episodes) by default. This creates unnecessary drain on the publishers' servers and consumes users' bandwidth.

In addition to user experience issues and slowing down the websites, if these guidelines are not followed, measurement companies may decide to discount ALL the traffic from these apps/sites because they cannot count true downloads or plays.

User Agent Structure

Recommendations for Device Platforms and App Developers / Publishers

Provide enough details in the user-agent header to allow it to be consistently differentiated from the user agent of other devices. Using the following pattern to build the user-agent will offer a consistent structure for all parties who consume the details:

```
<app name>/<app version><device info> <os name>/<os version><other info>
```

For example:

```
AppName/1.2.3 DeviceBrand DeviceModel OSName/1.2.3 LibName/1.2.3
```

Whenever possible the above user agent structure should be applied to both RSS feeds and audio files.

The Podcast Technical Working Group recommends that OS hosts should allow the user agent to be modifiable when using their libraries. The working group recommends that platforms be conservative in adding unnecessary information to the user-agent string, and in encoding practices. For example, it is best to refrain from injecting user or session IDs into the user-agent string.

We also recommend that platforms submit their user-agent header value to the IAB Spiders and Bots inclusion list so that it is not considered a bot and can be a signal used to determine the device information. If the app or platform does employ the use of bots to index content, it is recommended to specify a user-agent that is distinct from the application user-agent and includes the word "bot" to clearly identify its use case.

Automatically Triggered Downloads

At times some 3rd parties are identified as having implementations which do not conform to the Podcast Measurement Guidelines ultimately impacting the ability to capture accurate measurements if left unaddressed. This has resulted in the necessity to issue additional filtering guidance. In the future guidance found in this addendum may be appended or retracted as various 3rd parties make changes that require guidance updates.

Apple watchOS Duplicate Downloads: Filtering Guidance

In 2020 concerns were raised about Apple's watchOS Podcasts app. Downloads to an Apple Watch device appear to be duplicative of downloads to a user's iPhone after the watch performs its synching cycle. Additionally, members of the Podcast Technical Working Group have observed that the majority of watchOS downloads are automated downloads and not user initiated. The outcome of this could result in significantly inflated reporting of downloads and listeners.

Given the concerns for watchOS downloads, the Podcast Technical Working Group recommends that the following Apple Podcasts app watchOS user agent requests should be filtered out altogether up front and not be counted as unique downloads/listeners:

- UA's that begin with **atc/** and include **watchOS** (*for example atc/1.0 watchOS*)
- Or UA's that contain **(null)/(null) watchOS***

The IAB Tech Lab's participating members have agreed that by October 1, 2020, traffic from Apple Podcasts on watchOS devices will no longer be counted toward any totals reported for the podcasting industry.

Tech Lab has made efforts to work with Apple in order to support a means of differentiating automatic downloads from valid downloads coming from Apple Watch devices. Should Apple take steps to clearly distinguish those automatic downloads from user-initiated downloads, take other corrective actions, or make other changes that affect this guidance, it will be revised or retracted.