APPLICATION NOTE

INTERNET STREAMING SOLUTION WITH PROGRAM FEED USING AEQ PHOENIX AUDIOCODECS
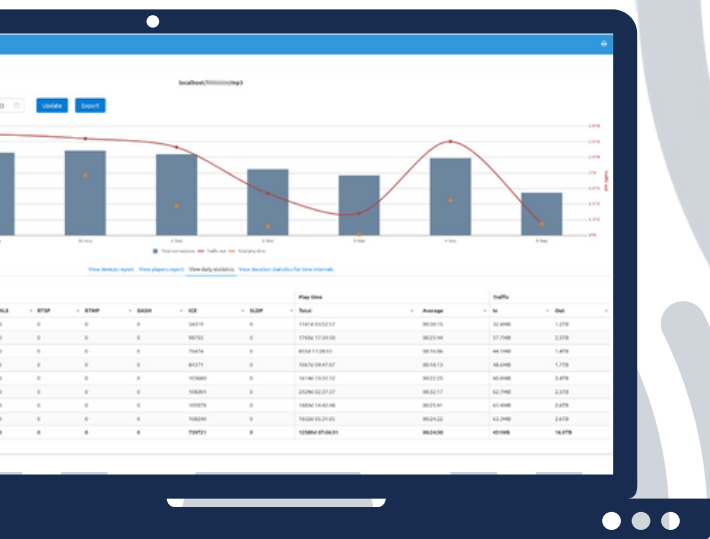
Table of contents

1. Introduction

More and more radio stations everyday broadcast their programs not only through radio waves but also by means of internet, and listeners are able to listen to them using an application or a simple Internet browser. The studio-generated audio must be consequently replicated in the network towards all the receivers. Application note 0-H explains how distribution is possible in a private or managed IP network using IGMP protocol. However, when distribution must be performed in the public Internet, this is not possible, as this Multicast protocol is not supported. This Application note intends to describe a solution to distribute audio through the Internet using an external replication service provider, where the audio program can be sent to, already encoded, using AEQ phoenix audio codecs.

2. Architecture

A cloud-based replication service takes an encoded audio stream and sends it to multiple destinations through the Internet. This service can be hired from providers that charge their fees according to the allowed maximum number of simultaneous listeners and, hence, the required bandwidth, as a copy of the stream must be sent to each one.

The server performing the replication can be within the station's network or outside, in the Internet. The advantage of the second alternative is that the station doesn't need to take care of the maintenance of the machine, and it doesn't need to hire a large Internet connection with a lot of bandwidth. Besides, the stream is sent already encoded to the server's "entry point", further reducing the requirements of that connection for the stream feed (audio input to the replication server).

This solution also allows for great flexibility, because the service level can be modified in order to expand the replication needs at a given moment (for example, when a large event takes place), without the need for additional investments in equipment or a larger bandwidth connection. The hired service mode can be temporarily upgraded (see table 1) while it is possible to return to the original level once the event is produced.

At the studio, only a dedicated high-quality audio encoder is required, which could desirably have different kinds of available inputs (analogue, digital or even Dante/AES-67). AEQ Phoenix codecs have been tested with several replication services thanks to the wide variety of encoding algorithms they offer and because they implement the RTP transmission standard, among other modes.

APPLICATION NOTE

Typically, replication servers require audio feeds in MP3 or MPEG-4 formats. The disadvantage of the first algorithm is that it doesn't stand very well to multiple re-coding (in case that's done in the server), degrading audio quality. On the other hand, MPEG-4 requires the payment of per-use fees or royalties. Even when AEQ codecs can optionally support several MPEG-4 modes (AAC-LC and AAC-LD families), there are more modern, efficient algorithms nowadays providing higher audio quality and besides they are free to use. That's why we have placed a bet on testing compatibility with replication servers which support the OPUS family of encoding algorithms.

One solution has been evaluated in particular: using the Spanish company Cires 21 (C21 Live Radio). CIRES support OPUS audio feeds. Several service levels are offered, mainly based on the maximum allowed number of simultaneous listeners and the bitrate of each replica (which ultimately determines its quality), although additional functionality is offered within each plan.
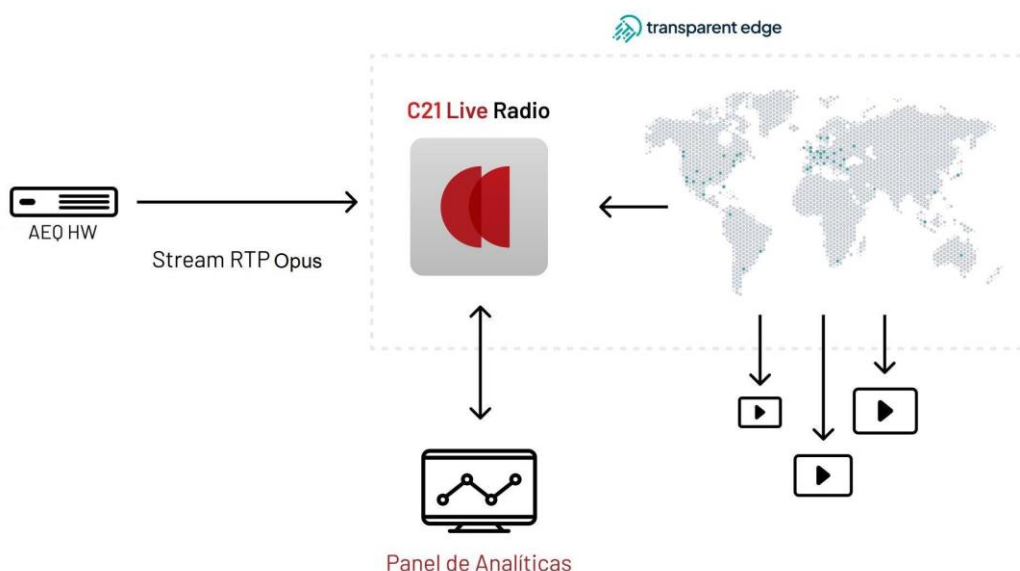
The selected plan has no influence in the configuration of the audio encoder, which stays within the radio station. The only required information includes the destination stream address and the encoding algorithm to use in the audio feed.

APPLICATION NOTE

The quality perceived by the users depends both on the encoding of the audio feed (which the audio codec performs and is sent once, so the bandwidth requirements are not an issue here) and the re-encoding performed by the replication server, which will use a moderate bitrate so the total data amount is not excessive, as it needs to be multiplied by the number of listeners. For compatibility reasons, Cires 21 offers the option of using MP3 as the final encoding algorithm, with 64 or 128 kbps.

Tests have been performed so far using bitrates with Cires 21 and the conclusion is that, depending on the kind of transmission, 64 kbps can be more than enough if voice only or a mono signal is sent, but if we intend to transmit stereo music, our recommendation is to use at least 128 kbps per stream in order to get good quality.

As an example, these are the plans offered by AEQ in collaboration with Cires 21 using Phoenix audiocodecs, and their distinctive characteristics:

| AEQ + C21 LIVE RADIO			
	Basic	Advanced	Specialized
Service Level	Up to 500 simultaneous users / 5TB monthly traffic	Up to 8000 simultaneous users / 80TB monthly traffic	Up to 25000 simultaneous users / 250TB monthly traffic
RTP audio feed	✓	✓	✓
Shoutcast/Icecast output formats	✓	✓	✓
HLS/MPEG-DASH output formats		✓	✓
VAST Advertisement insertion		✓	✓
Basic Statistics	✓	✓	✓
Advanced Stats and real-time broadcast access registers		✓	✓

Table 1: Service Levels.

In case that you need a different configuration, output format, etc. please contact your AEQ sales representative.

3. Audio codec configuration

When working with Cires21, the stream input must be sent according to RTP Standard. In order to stablish an RTP stream, we can use a single-channel **Phoenix Mercury** audiocodec.

First, we need to define the proper encoding algorithm required to feed the server, for example: OPUS MUSIC 128kbps stereo.
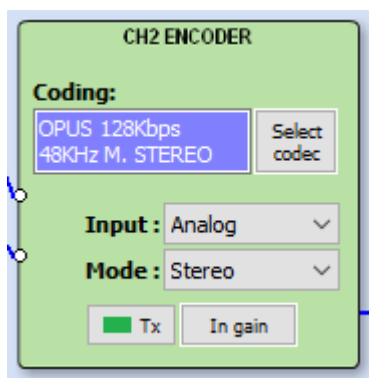
Next, we need to have in mind that transmission is unidirectional, so we need to be aware that the auto-hang-up option (when RTP input is lost) must be de-activated in the codec so it doesn't interrupt the streaming after some seconds without receiving anything from the replication server, as this will be always the case.
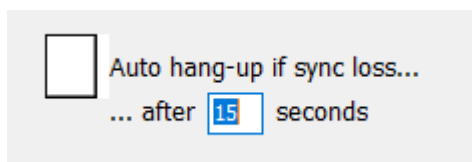
It is also convenient to activate the Permanent Call option, located in the same dialog box, so the unit re-establishes the streaming transmission after a reboot or power interruption.

Once these features have been configured, we only need to make the call to the entry point defined by Cires 21.

Only the server IP address and port need to be specified, and these values are provided by Cires 21, i.e.:

149.202.176.155:20012

NOTE: In some cases, the IP address is not numeric but it has a domain name format. AEQ audiocodecs also support address resolution if a DNS server is specified in the Ethernet configuration properly, so server addresses like this one, which is equivalent, can also be used:
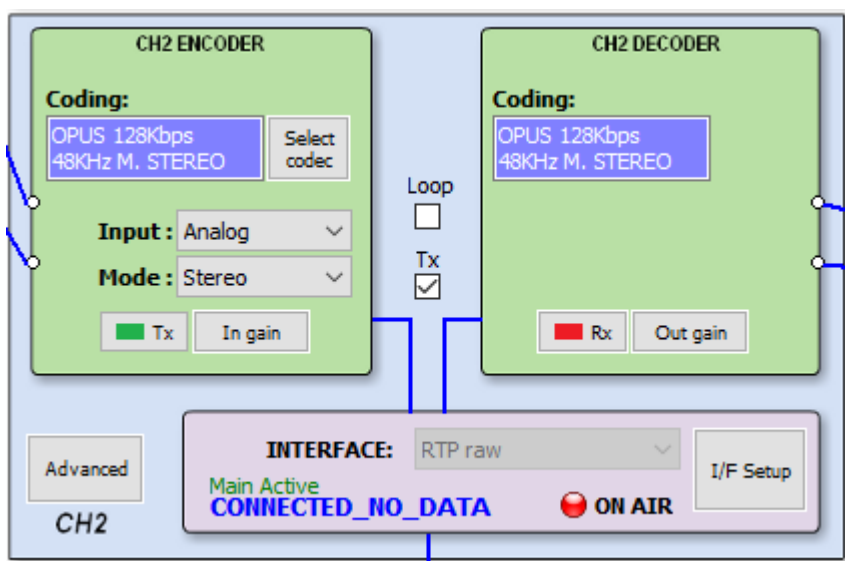
primary.democdnradio.cires21.com:20012

Using ControlPhoenix, click on the green Call button and fill the correct address in (or, alternatively, use a previously defined entry in the Call book).
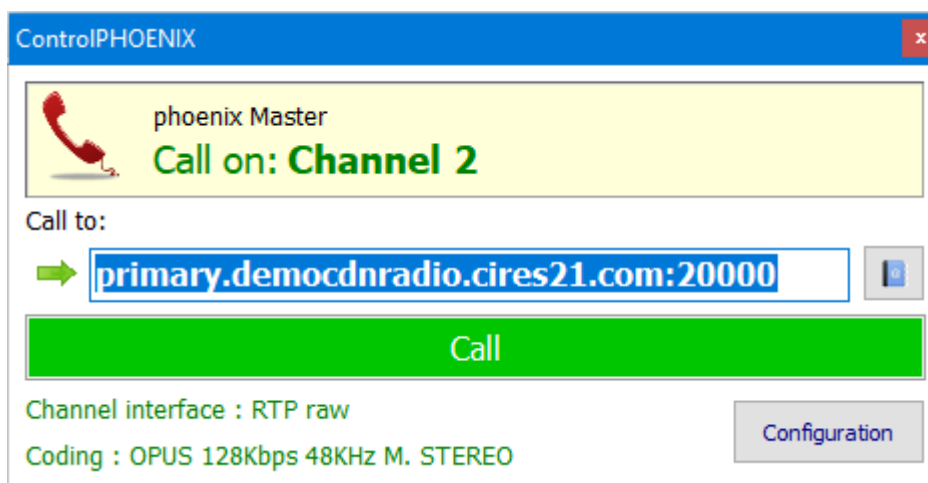
Once the call is made, please observe that the codec channel status will always be CONNECTED_NO_DATA. This is correct, as no return traffic is received from the server.

4. Receiving the stream

In order to listen to the stream, users have two alternatives:

On one hand, it is possible to directly use a URL in order to listen to the stream using a Web browser (which must include the proper codecs).

On the other hand, it is also possible to use a compatible player application. In this case, an address is used to open the stream. Some examples of these players are VLC, FFplay, VideoJS or Shaka-Player. In all of them, an option is available to open a network location.

Cires21 demonstration platform allows us to send up to 6 different streams. Each entry point has a different port assigned, and in order to listen to each stream, the URL format will be:

When using a web browser to play the stream:

Cires21 streams can be listened to using a link with this format:

http://cires21.liveradio-edge.transparentedge.io:8081/**<stream>**/output/icecast.audio

> (where **stream** is an identifier of the selected channel and *must be substituted by "ll-stream001", " ll-stream002"… "ll-stream006")*

Listening to the stream using a dedicated player:

All programs offer an option to open a network location or similar.
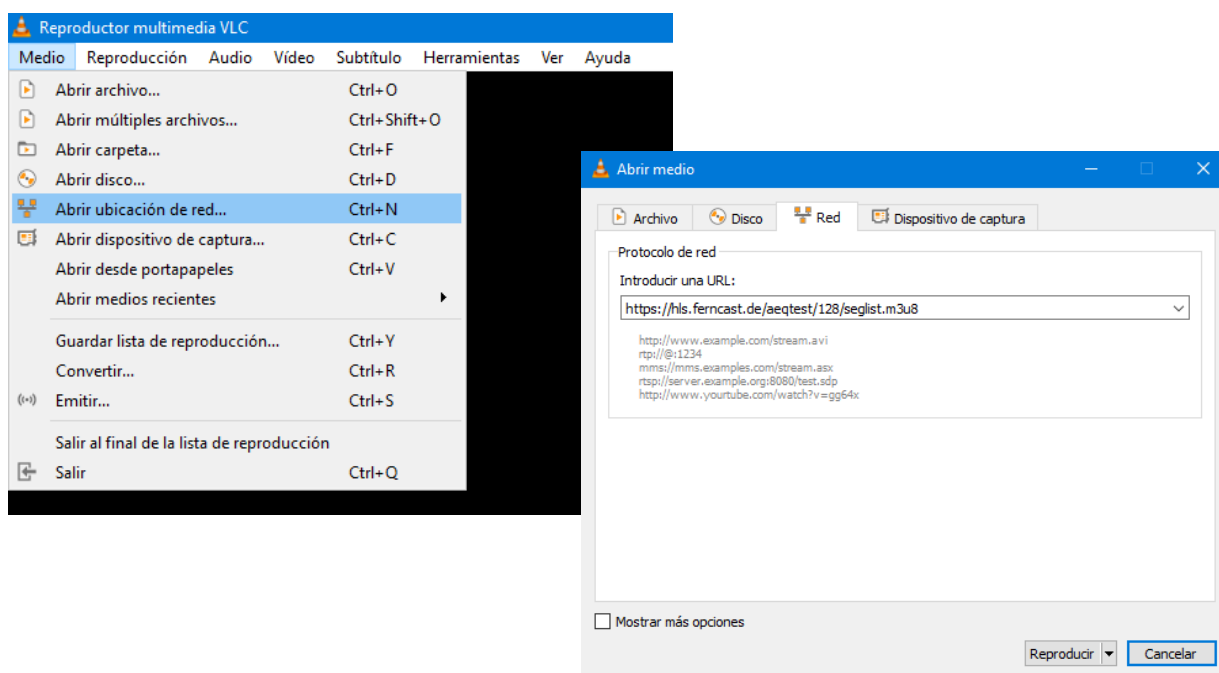
Type in the URL provided by Cires 21 there, for example:

https://cires21.liveradio-edge.transparentedge.io/**<stream>**/output/playlist.m3u8

(where <stream> must be substituted by "ll-stream001", " ll-stream002"… "ll-stream006")

5. Delay

Every internet-based stream replication system introduces significant delays. This delay is contributed by reception in the server, decoding, buffering, re-encoding and distribution through all the Internet backbone, but, mainly, by the required large buffers in the receiving end. This buffer's mission is to absorb large delay variations (jitter) among the different audio packets which is produced in such a large network, so all of them can be received, buffered and re-ordered before their turn to be played back arrives.

Using an internet browser, end-to-end delays starting in 8 seconds can be obtained. VLC players or similar tend to use larger buffers so delays up to 30 seconds are usual.

This unavoidable delay must be had into account when selecting a system of this kind, as it could be excessive for certain program types such as live sports transmissions, while it is not important for music or chats broadcasting.

6. Redundancy

Cires 21 offers an optional redundant-servers service. That is, two servers are offered which will create the same streams to the Internet, each one with its own encoded audio inlet. If one of them fails, the streams will automatically be generated by the other one, so the service is not interrupted. This transition is, however, not seamless.
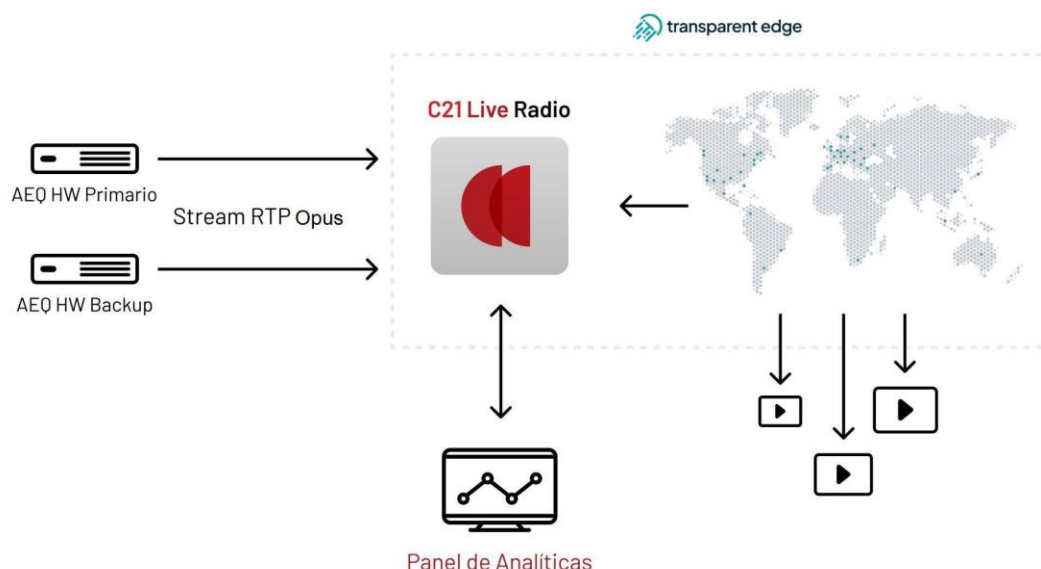
This also allows for the implementation of ISP redundancy: we can have two separate internet accesses in the studio, and use two **Phoenix Mercury** audiocodecs, each one connected to an ISP. We must input the same audio source to both of them (in analogue or digital format). Each one will call to a different destination address, as specified by Cires 21. For example:

MERCURY 1 → primary.democdnradio.cires21.com:20012

MERCURY 2 → backup.democdnradio.cires21.com:20012

This way, if one of the Internet accesses is down, the corresponding replication server would stop receiving traffic, so the system would automatically generate the replicated streams using the other server.

This can also be implemented using a single **Phoenix Venus 4 o Venus4+,** each channel using a different Ethernet port to each ISP provider. In this case, feeding the same audio signal to both channels is required (which is particularly easy when using Dante input). The calls to establish would be these:

VENUS CH1 → primary.democdnradio.cires21.com:20012

VENUS CH2 → backup.democdnradio.cires21.com:20012

NOTE: The system always tries to minimize changes, so if the fallen server or link recovers, it won't generate the streams until the complementary one also falls.

NOTE: The transition from Primary to Backup (or the other way around) is not "seamless". That is, if the user is listening to a broadcasting through Internet and one of this server failovers happen, the audio will be interrupted and the user will need to reload the browser

page or reopen the playback link in the application. This change can take around 10 seconds, approx.

7. Monitoring

Cires 21 offers a web access where, once logged in using the credentials provided by them, we can have access to connection statistics, check the number of connected users, etc.

In order to gain access, type the following URL in your browser:

<div align="center">

http://c21liveradio.cires21.com/

</div>

Once logged in, the following information is available in the BASIC reports plan (see available plans table).

7.1. Main Dashboard

This is the service's advanced stats main screen, and it is accessed right after logging in.

It provides information grouped in "Slices" or "views". Each "Slice" provides access to the consumptions or one or several streams distributed by the platform. The dashboard is divided into two sections:
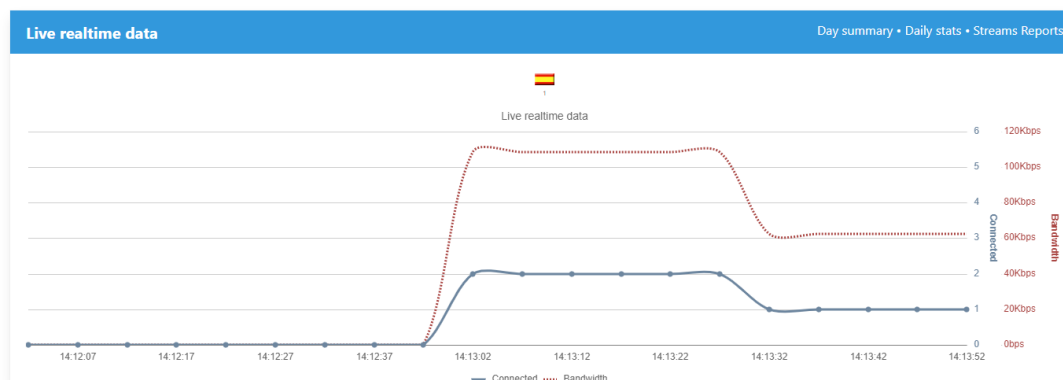
Live real-time data

It shows real time data about the data consumption during the last minute associated to the stream or streams that correspond to the selected "Slice".

The data we´ll be able to visualize is:

- Number of connected users.

- Consumed bandwidth.

- Localization of users per country of origin including the number of users in each location.

This day's Retrospective data

It shows retrospective data about the data consumption associated to the stream or streams that correspond to the selected "Slice". It allows for the representation of consumption data on different time ranges:

DAY: corresponds to the last 24 hours.

WEEK: Last week of service.

MONTH: Last month of service.

YEAR: Last year of service.

The data set we´ll be able to visualize is:

- Number of users per time range.

- Bandwidth consumed per time range (Mbps).

- Traffic distributed per time range (Gb).

7.2. Reporting Menu

The statistics system also allows us to consult retrospective data about consumption. The available reports are:

**Dashboard:** returns to the main dashboard described on section 7.1.

Day Summary: shows detailed consumption data for a specific day of service. The data set that the system makes available to the customer is:

- Total number of connections.

- Total consumption time of the stream or streams.

- Average time of consumption.

- Traffic distributed by the service.

- Maximum peak of bandwidth consumed by the distribution of the service.
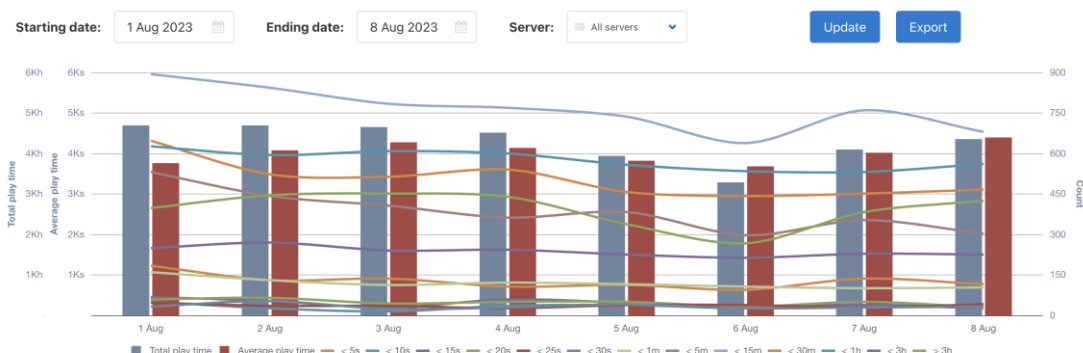
Daily stats: shows the consumption data of each day between two configurable dates.
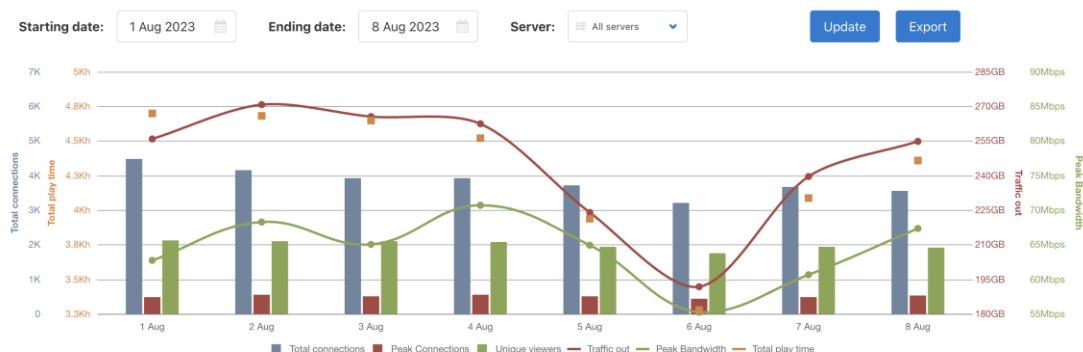
The data set that the system makes available to the customer through this report is:

- Total number of connections per day, as well as the total.

- Total time consumed for the stream or streams per day, as well as the total.

- Average consumption time per day, as well as the total.

- Traffic distributed by the service per day, as well as the total.

- Maximum peak of bandwidth consumed by the distribution of the service.

Duration stats: shows each day's consumption data between two configurable dates. The data that the system makes available to the customer through this report is related to the duration of the established connections.

Devices report: shows the consumption data of each day between two configurable dates. The data that the system makes available to the client through this report is related to the device from which the consumption is done by the final user.

Players report: shows the consumption data between two configurable dates. The data that the system makes available to the client through this report is related to the player from which the consumption is done by the final user.

Geo stats: shows the data of the percentages of usage depending on the geographic region, representing it on a map:


Streams report: Provides the number of connections per stream in real time, grouped over the view.

7.3. Advanced statistics

The advanced statistics system, available in the Advanced and Specialized plans, offers, apart from the already described statistics, other data:

- **Deep stats**: This report offers a complete description of the distribution of the content grouped on a view or slice. Every live transmission has a complete set of daily statistics; this kind of report allows access to the data of each available stream for its later interpretation. It even allows you to generate comparative reports between streams.
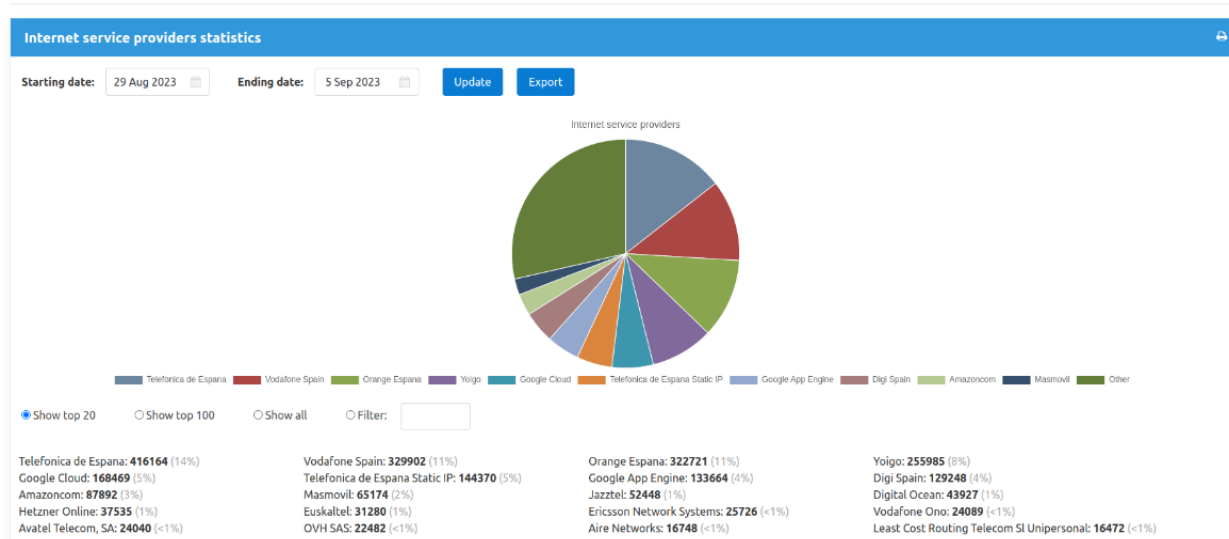
APPLICATION NOTE
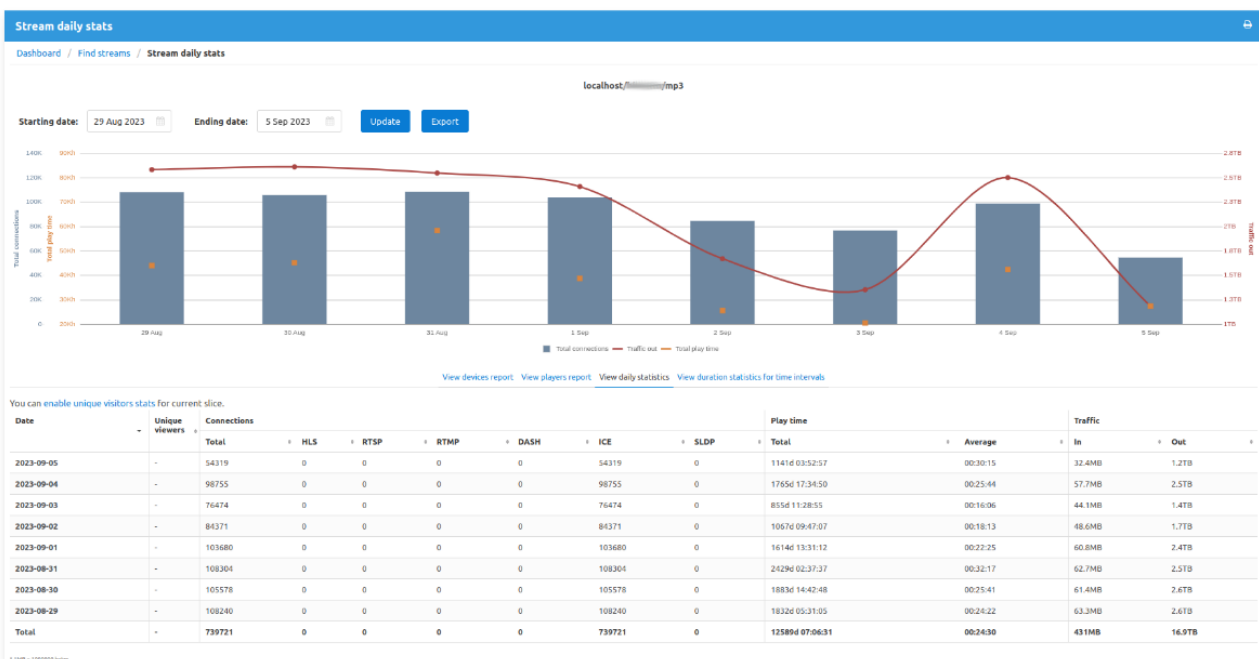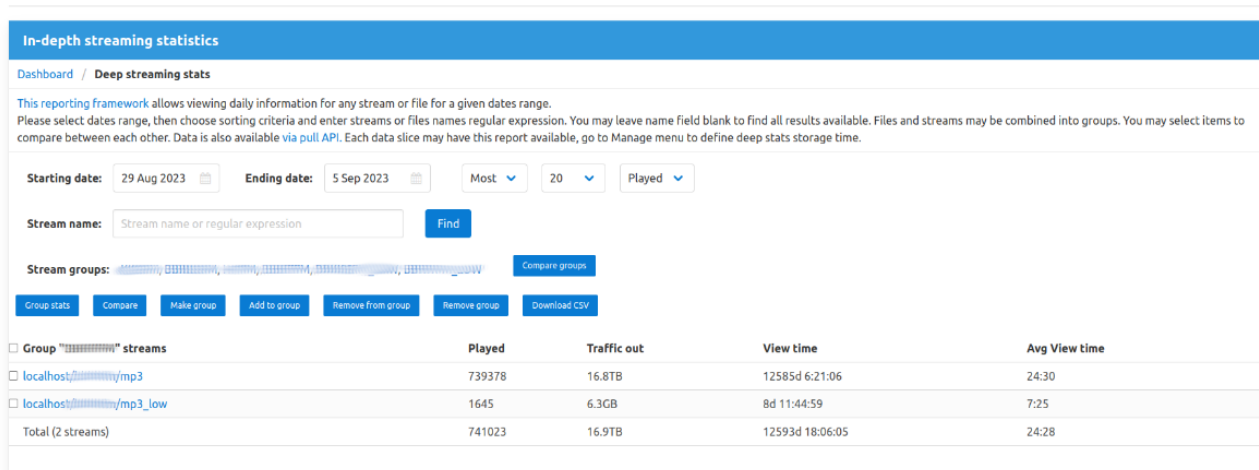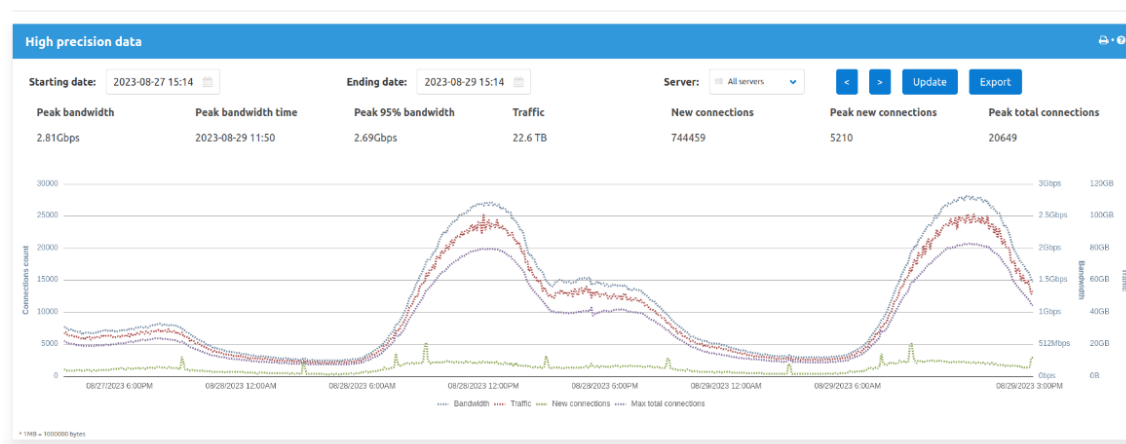
APPLICATION NOTE
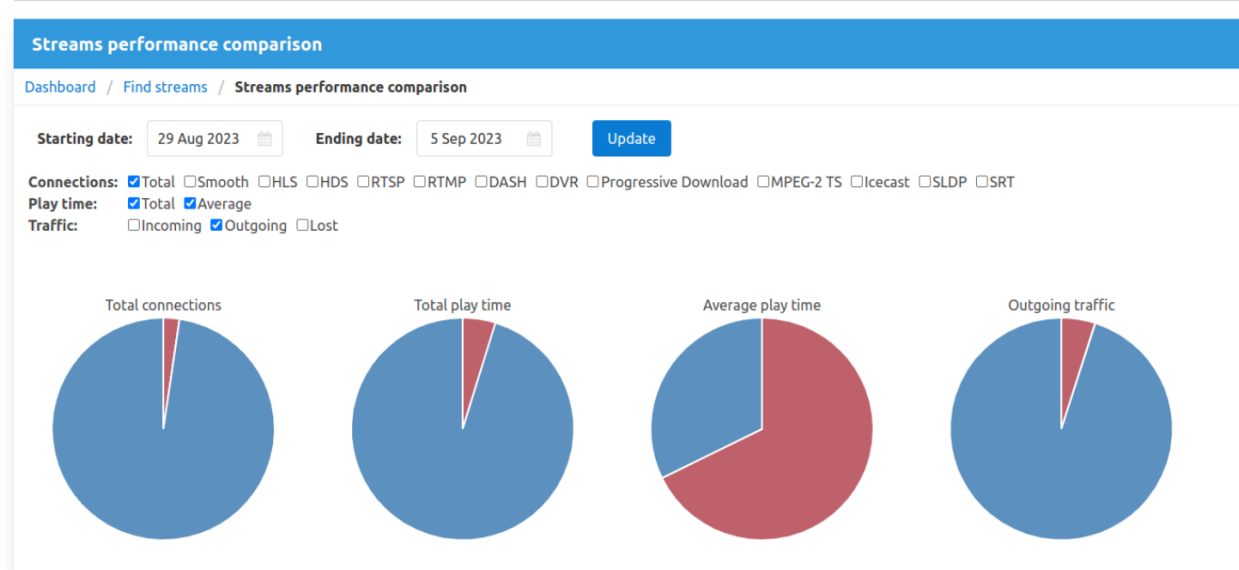
- **ISP stats**: Report that collects statistics from Internet service providers for the selected time period.

- **High precision data**: Report that provides the following data with a granularity of 5 minutes over a stablished time period of 48 hours, maximum:
    - Maximum bandwidth
    - Bandwidth peak hour
    - 95% of maximum bandwidth
    - Total transferred traffic
    - Total connection count

Additionally, mechanisms are provided for the emission of access logs towards third party systems.

AEQ, S.A.

APPLICATION NOTE: R+D DEPARTMENT,

AEQ MADRID (SPAIN)