

Mcaster Manual

Sistema modular profesional para el procesamiento de señales de TV

Table of contents

1. Productos	3
2. Mcaster	4
2.1 Características funcionales	5
3. Instalación	7
3.1 Instalación	7
3.2 Inicio rápido	9
3.3 Administradores	11
4. Captura de señal	20
4.1 Captura de multicast y MPTS	20
4.2 Captura por SRT y RIST	22
4.3 Publicación SRT	24
4.4 Recepción de ST 2110	28
4.5 Reserva de entradas	36
4.6 Errores de entrada	38
5. Procesamiento	40
5.1 Salida MPEG-TS	40
5.2 CBR para DVB	42
5.3 Subtítulos y teletexto	44
6. Entrega	46
6.1 Entrega a la red	46
6.2 Entrega por SRT	48
6.3 La copia OTT	49
6.4 Monitorización de entrega	51
7. Archivo	56
7.1 El archivo	56
8. Control de calidad	58
8.1 Contadores de transporte (TR 101 290)	58
8.2 Monitorización	60

1. Productos

2. Mcaster

2.0.1 Mcaster

Mcaster es la cabecera de Flussonic. La señal le llega de un receptor de satélite, de una red vecina, de un socio o de su propio playout; Mcaster la lleva a la forma que acepta la red de entrega y **la hace llegar al destino** — al multicast de una red de cable, a un modulador, a otra ciudad por SRT, a un emplazamiento de reserva.

La diferencia clave con [Catena](#): Catena termina en el espectador, Mcaster termina en el punto de entrega de la señal. Más allá de ese punto hay un espectador, pero ese espectador no es el cliente de la cabecera. De ahí el tema de este manual: un flujo de transporte con sus propios PSI/SI, un bitrate constante, contadores de transporte y prueba de entrega por cada destino.

Qué puede poner en marcha

- **Recepción desde satélite y desde una red ajena.** El programa se escoge de un troncal MPTS, y el propio troncal se recibe una sola vez, sin importar cuántos programas se tomen de él.
- **Transporte entre emplazamientos.** SRT por internet abierto con cifrado y recuperación de pérdidas; UDP/RTP dentro de la propia red.
- **Preparación de la señal para el receptor.** PNR, PID y SDT propios por destino; un bitrate constante para el modulador; teletexto y subtítulos que lleguen vivos a la salida.
- **Entrega comprobada.** Por cada destino: estado con un motivo de fallo con nombre, velocidad, bytes, reconexiones, retransmisiones.
- **Un archivo en el propio emplazamiento.** La grabación directamente en la cabecera — para el día en que alguien pregunte qué exactamente se emitió a las 03:40.

La lista completa está en la página [Características funcionales](#).

Cómo está construido

Mcaster es una interfaz y una licencia sobre el motor de streaming [Sapsan](#); la compilación es la misma. De ahí se siguen tres consecuencias prácticas.

- **Todo se configura desde el panel central.** Los streams, las máquinas, el procesamiento y los destinos de entrega viven en una sola interfaz web; no hace falta editar archivos en los servidores.
- **Un proceso, una estación.** La instalación mínima es una sola máquina: motor, panel y API en un único proceso.
- **Varias máquinas, una interfaz.** Cuando hay más de una estación, se conectan a un plano de control y el operador las ve en el mismo panel: la distribución de los streams entre máquinas, un archivo compartido, estadísticas compartidas. Una máquina conectada no tiene ajustes propios — los recibe del plano de control.

Un stream dentro de Mcaster es un solo servicio: tiene entradas (principal y reserva), procesamiento y salidas. Todo lo que se describe en este manual es una propiedad del stream.

Por dónde empezar

- [Instalación](#) — del paquete a un proceso en marcha.
- [Inicio rápido](#) — capturar un multicast y devolverlo a la red, con una comprobación en cada paso.
- [Captura de multicast y MPTS](#) — lo primero que hace una estación.
- [Monitorización de la entrega](#) — cómo asegurarse de que la señal llegó.

2.1 Características funcionales

La lista de funciones de la cabecera Mcaster. Cada función está descrita en una página de este manual, con el enlace al lado.

2.1.1 Captura de señal

- MPEG-TS por UDP — unicast y multicast, IPv4 e IPv6, con suscripción al grupo y selección de fuente (SSM) — [más](#).
- Selección de programa de un troncal MPTS por número (`program=`); el troncal se recibe una sola vez para todos sus programas, sin importar cuántos streams lo interpreten — [más](#).
- MPEG-TS por HTTP, incluida la transferencia por chunks; una conexión rota y una fuente silenciosa se distinguen — [más](#).
- Captura por SRT: un puerto dedicado por stream, cifrado AES, `streamid`, un presupuesto de recuperación de pérdidas ajustable — [más](#).
- Captura por RTMP y HLS — para fuentes que llegan del mundo OTT.
- Códecs de entrada: video H264, HEVC, MPEG-2; audio AAC, AC3, EAC3, MPEG-2; teletexto DVB, subtítulos DVB, subtítulos CEA-608/708, metadatos KLV, marcadores SCTE-35.
- Redundancia de entrada: varias fuentes por stream, conmutación automática según la salud de la fuente, retorno a la principal tras una ventana limpia — [más](#).
- Detalle causal de los errores de entrada y resaltado de los minutos sin señal — [más](#).

2.1.2 Procesamiento

- Formación del flujo de transporte de salida: PAT/PMT/SDT, número de programa, PID explícitos de las pistas, nombres de servicio y de proveedor, `transport_stream_id` y `original_network_id` — [más](#).
- Tablas de servicio propias por destino: la misma señal llega a distintos receptores con distintos PNR y PID, sin un segundo multiplexor — [más](#).
- Bitrate constante (CBR) con relleno de paquetes nulos, PCR sin deriva y consideración del modelo de búfer del decodificador (HRD) — [más](#).
- Transcodificación: video H264, HEVC y AV1; audio AAC y Opus; desentrelazado, escalado, control de GOP — [más](#).
- Teletexto y subtítulos: paso a la salida tal como llegaron, extracción a una pista de texto, o ambas representaciones a la vez — [más](#).

2.1.3 Entrega

- Envío a la red por UDP: multicast y unicast, control de TTL, selección de interfaz de salida y de dirección de enlace — [más](#).
- Encapsulación RTP según RFC 2250 / SMPTE ST 2022-2 — la misma salida, el esquema `rtp://` — [más](#).
- Envío por SRT a un destino fijo con cifrado y `streamid`; servicio SRT desde el propio puerto del stream — [más](#).
- Envío por RTMP a plataformas y CDN, incluidos HEVC y AV1 (enhanced RTMP) — [más](#).
- Varios destinos por stream a la vez; los destinos con ajustes idénticos comparten un único flujo idéntico byte a byte — [más](#).
- Reanimación automática de un destino muerto sin intervención manual — [más](#).
- Una copia OTT de la señal: HLS, LL-HLS y DASH desde la misma estación — [más](#).
- Cifrado de la copia OTT: Common Encryption `cbs` para HLS y DASH, la clave desde los ajustes o desde su servidor por SPEKE, señalización para Widevine, PlayReady, FairPlay y ClearKey; los pushes salen sin cifrar — [más](#).

2.1.4 Archivo

- Grabación en un almacenamiento append-only repartido en varios discos, retención por profundidad y por tamaño — [más](#).
- Reproducción del archivo por tiempo absoluto, rewind, unión sin costura con el directo, vistas previas JPEG — [más](#).

2.1.5 Control de calidad

- Contadores de transporte por cada PID: violaciones del continuity counter, TEI, paquetes cifrados, errores CRC de PSI, PES dañados, jitter de PCR frente a los umbrales del estándar, déficits del búfer del decodificador — [más](#).

- Contadores de entrega por destino: estado con motivo, velocidad, bytes, errores, reconexiones, retransmisiones — [más](#).
- Detalle causal de los errores de entrada: los defectos de medios y los fallos de fuente van separados, cada causa es su propia línea en el gráfico — [más](#).
- Un servidor Prometheus integrado como fuente de datos lista para Grafana, un endpoint de scrape para su propia TSDB, telemetría hacia la monitorización en la nube Retroview — [más](#).

2.1.6 Gestión

- Un panel central: streams, máquinas, distribución, archivo y sesiones en una sola interfaz web.
- Las ediciones se aplican al vuelo, sin interrumpir la difusión.
- Un solo panel para todo el clúster: una máquina conectada recibe sus ajustes del plano de control.
- Una API de gestión para la integración con sistemas de aprovisionamiento de terceros.
- Un rechazo estructurado de un ajuste no válido: 422 con el campo culpable señalado, y no una distorsión silenciosa de la configuración.

3. Instalación

3.1 Instalación

Mcaster se instala en una máquina y en ella misma funciona: motor, panel de control y API en un único proceso. Esta página trata de lo que deja el paquete, de dónde viven los secretos y de qué se abre hacia fuera. El camino de una máquina limpia a una señal en la red está en el [Inicio rápido](#).

3.1.1 El paquete

Mcaster se distribuye como paquete deb `mcaster` desde el repositorio de Flussonic:

```
curl -L http://apt.flussonic.com/repo/master/dev.key > /etc/apt/trusted.gpg.d/dev.gpg
echo "deb http://apt.flussonic.com/branch/sapsan/master repo/" > /etc/apt/sources.list.d/flussonic.list
apt update
apt install --install-recommends mcaster
```

El paquete se hace cargo de toda la superficie de la instalación:

Qué	Dónde
Unidad	<code>mcaster.service</code>
Ajustes del proceso	<code>/etc/mcaster/</code> — lo que el propio proceso necesita al arrancar: el puerto del panel, la dirección de la base de datos, la licencia
Secretos	<code>/etc/mcaster/secrets.env</code> (propietario root, permisos 0600)
Licencia	<code>/etc/mcaster/license.txt</code>
Estado	<code>/var/lib/mcaster</code> — identidad de la máquina, caché de activación, claves
Interfaz	<code>/usr/share/mcaster/ui</code>

El paquete instala automáticamente el motor de streaming exactamente en su propia versión, junto con PostgreSQL. La unidad base `sapsan.service` se retira de una máquina así: debe funcionar exactamente un proceso. Los productos son mutuamente excluyentes — Mcaster y Catena no comparten máquina.

Note

La actualización del paquete no reinicia una unidad en marcha. La máquina pasa al nuevo par «binario + interfaz» con un `systemctl restart mcaster` manual — así el reinicio sigue siendo una decisión del operador y no un efecto secundario de `apt upgrade`.

3.1.2 Licencia

Sin una clave válida el proceso no arranca. La clave se solicita durante la instalación y se escribe en `/etc/mcaster/license.txt`; para despliegues desatendidos se introduce mediante `debconf-set-selections` o la variable `LICENSE_KEY`.

La clave se emite **para un producto**: la máquina se declara como `mcaster`, y una clave emitida para otro producto no le servirá — no se concede activación, y eso se ve en el log de inmediato, y no a través de una función que no funciona.

Dejar la pregunta en blanco está permitido: el paquete se instala, pero el servicio queda en `failed`. Ponga la clave en el mismo archivo y ejecute `systemctl start mcaster`.

3.1.3 PostgreSQL

Mcaster es una instalación de control y guarda su estado (streams, máquinas, estadísticas, registro) en PostgreSQL. El script de postinstalación crea el rol y la base de datos `mcaster`, genera las contraseñas de la base de datos y del administrador y las escribe en `/etc/mcaster/secrets.env`:

```
DATABASE_URL=postgres://mcaster:...@127.0.0.1:5432/mcaster
EDIT_AUTH_PASSWORD=...
```

La actualización del paquete no rota los secretos. Para pasar la base de datos a un clúster gestionado con copias de seguridad, cambie `DATABASE_URL` y reinicie el servicio — Mcaster aplica su esquema por sí mismo al arrancar, no hay un paso de migración aparte.

3.1.4 Puertos

Recién instalado, se levanta el puerto HTTP 80 — un valor *inicial* que solo se mantiene hasta que los listeners se fijan en el panel. A partir de ahí los puertos pertenecen a los ajustes de la máquina y ahí mismo se editan.

Puerto	Quién lo fija	Para qué
HTTP (inicialmente 80)	<code>INITIAL_HTTP_PORT</code> , luego los ajustes de la máquina	Panel, API, salida por HTTP
Entradas UDP	El puerto de cada fuente multicast	Captura de MPEG-TS
SRT (UDP)	El puerto de servicio SRT del stream o el listener compartido	Captura y salida por SRT
RTMP (TCP)	Un listener en los ajustes de la máquina	Captura y salida por RTMP

Los destinos de salida no ocupan puertos: un push abre su propio socket.

El multicast exige atención a las rutas. La captura es una suscripción al grupo (IGMP/MLD) en la interfaz por donde llega el grupo; la salida es la [elección de la interfaz de salida y del TTL](#) en el propio destino. En una máquina con varias tarjetas de red ambas elecciones se hacen de forma explícita; de lo contrario, los paquetes se van a la red equivocada.

3.1.5 Ejecución en Docker

La imagen de producto es `flussonic/mcaster`: el runtime compartido más la interfaz de Mcaster y el tag de producto. Necesita PostgreSQL, una clave de licencia y un directorio de estado persistente:

```
docker run -d --name mcaster \
-e DATABASE_URL=postgres://mcaster:mcaster@db:5432/mcaster \
-e EDIT_AUTH_LOGIN=admin -e EDIT_AUTH_PASSWORD=secret \
-e LICENSE_KEY="$LICENSE_KEY" \
-p 8081:80 \
-v mcaster-state:/var/lib/sapsan \
flussonic/mcaster:latest
```

Los puertos de los protocolos de medios que activen los ajustes de los streams también se publican aquí. El directorio de estado debe sobrevivir a los reinicios: contiene la identidad de la máquina y la activación.

3.1.6 Comprobación

```
systemctl status mcaster
journalctl -u mcaster -n 50
```

Lo siguiente es el [Inicio rápido](#): entrar en el panel, el primer multicast capturado y el primer destino de entrega.

3.2 Inicio rápido

Esta página lleva una máquina limpia hasta una señal recibida de la red y devuelta a la red — la razón de ser de una cabecera. Aquí no se configuran el clúster, el archivo ni la transcodificación: eso llega después.

3.2.1 Qué necesita

- Una máquina con Ubuntu 24.04 y acceso root, en el mismo segmento de red por donde circula el multicast.
- Acceso a internet — la licencia se activa en línea.
- Una clave de licencia de su cuenta en my.flussonic.com. La propia instalación se la pedirá.
- Un grupo multicast vivo con MPEG-TS — por ejemplo `239.255.1.77:5500`. Si no lo hay, empiece con la fuente sintética: ejercita todo el recorrido, salvo la captura.

3.2.2 Instale

```
curl -L http://apt.flussonic.com/repo/master/dev.key > /etc/apt/trusted.gpg.d/dev.gpg
echo "deb http://apt.flussonic.com/branch/sapsan/master repo/" > /etc/apt/sources.list.d/flussonic.list
apt update
apt install --install-recommends mcaster
```

Con un solo paquete basta: instala automáticamente el motor en su propia versión y PostgreSQL, crea la base de datos, genera las contraseñas y arranca `mcaster.service`. `--install-recommends` mantiene funcionando el transcodificador — llega como dependencia recomendada.

Qué es exactamente lo que instala el paquete está en la página [Instalación](#).

3.2.3 Entre en el panel

El paquete escribió la contraseña del administrador en el archivo de secretos:

```
grep EDIT_AUTH_PASSWORD /etc/mcaster/secrets.env
```

Abra `http://<server-address>/` e inicie sesión como `admin` con esa contraseña.

3.2.4 Capture el multicast

Vaya a **Streams**, introduzca el nombre `tv1` y pulse **Crear**. El stream se abre en la pestaña **Fuentes**.

Pulse **+ Añadir fuente**, elija el protocolo **UDP MPEGTS** y rellene:

- **Host** — la dirección del grupo, `239.255.1.77`;
- **Puerto** — `5500`;
- **Tiempo de espera del par (ms)** — cuánto silencio en el grupo tolerar antes de reiniciar la captura; un campo vacío significa el valor por defecto.

Pulse **Guardar**. La estación se suscribe al grupo y en pocos segundos el stream se pone verde: aparecen un bitrate y una lista de pistas.

Si el grupo transporta un multiplex y no un único servicio, ponga el número en el campo **Programa (PNR)** — véase [Captura de multicast y MPTS](#).

3.2.5 Envíelo a la red

Abra la pestaña **Egreso** y pulse **+ Añadir push** en el bloque **Pushes**:

- **Nombre** — `headend-out`. El nombre es la identidad del destino: por él se encuentra el push en los ajustes, en las estadísticas y en las métricas.
- **Protocolo** — **MPEG-TS/UDP**.
- **Host y Puerto** — el grupo al que se entrega la señal, por ejemplo `239.1.1.10` y `5000`.
- **TTL de multicast e Interfaz de salida** — se rellenan cuando la máquina tiene varias tarjetas de red o el paquete tiene que sobrevivir a un router.

Pulse **Guardar**. Bajo los ajustes del push aparece una línea de estadística en vivo: **Enviando**, la velocidad en kbit/s, los errores por segundo, las reconexiones.

3.2.6 Compruebe qué llegó

Desde una máquina situada en la red de destino, suscríbase al grupo y mire lo que corre en él: primero si los paquetes llegan siquiera (`tcpdump -i <iface> host 239.1.1.10`), luego si su receptor o analizador de flujo de transporte interpreta un programa con las mismas pistas que muestra la tarjeta del stream.

Qué significan el número de programa y los PID en su salida, y cómo fijarlos, está en la página [MPEG-TS de salida](#).

La segunda fuente de verdad son las estadísticas de la propia estación: la línea bajo el push responde «estamos enviando», el receptor responde «llegó». El desacuerdo entre esas dos respuestas es exactamente el aspecto que tiene un problema de red. Los contadores se explican en la página [Monitorización de la entrega](#).

3.2.7 Si algo salió mal

```
systemctl status mcaster
journalctl -u mcaster -n 50
```

- **El servicio no está en marcha y el log habla de la licencia.** La clave no se introdujo durante la instalación. Póngala en `/etc/mcaster/license.txt` y ejecute `systemctl start mcaster`.
- **El servicio no está en marcha y el log habla de la base de datos.** PostgreSQL no es accesible: escriba un `DATABASE_URL` que funcione en `/etc/mcaster/secrets.env` y reinicie el servicio.
- **El stream está en rojo, no hay bitrate.** Los paquetes del grupo no llegan a la máquina: una VLAN distinta, una petición IGMP que el conmutador no deja pasar, un firewall que corta el UDP, un grupo o un puerto equivocados. Compruebe desde esta misma máquina — es ella la que se suscribe.
- **Hay bitrate, pero no hay pistas.** El stream llega, pero no es ese servicio: el grupo transporta un multiplex y hace falta un número de programa.
- **El push dice «Enviando» y el receptor no ve nada.** Los paquetes salen a la red equivocada o mueren en un router: fije la **Interfaz de salida** y suba el **TTL de multicast**.
- **El push está en estado «Error».** El motivo está escrito ahí mismo, junto al estado: conexión rechazada, timeout, rechazo del receptor.

3.2.8 Qué viene después

- [Añadir una entrada de reserva](#) — una segunda fuente y conmutación automática.
- [Fijar el PNR, los PID y el SDT](#) — para que el receptor vea el servicio con el número correcto.
- [Obtener un CBR](#) — cuando la señal va a un modulador.
- [Configurar la monitorización](#) — alertas para una entrega que se detuvo.

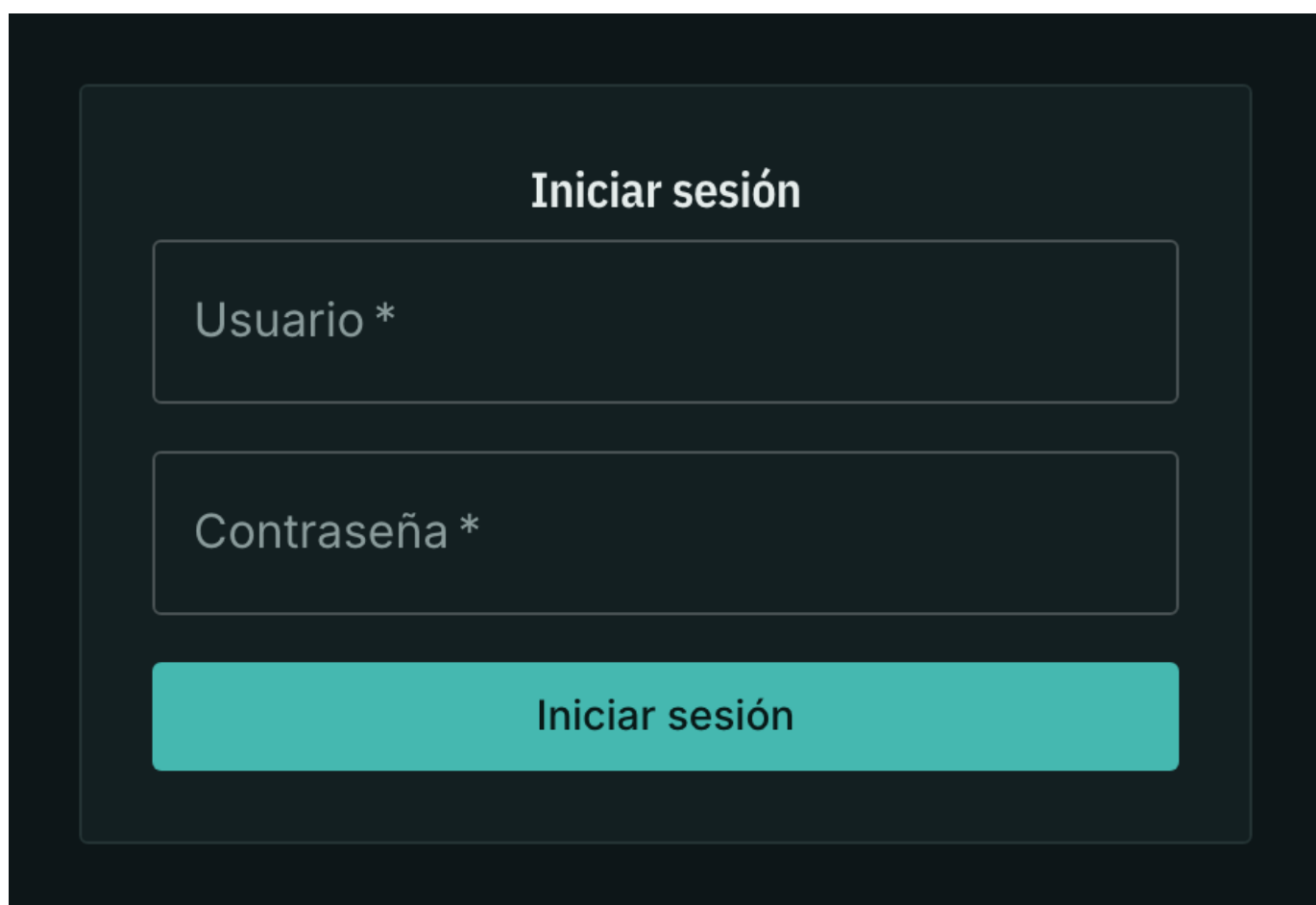
3.3 Administradores

El acceso al panel de Mcaster se rige por cuentas nominativas: todo el que abre el panel tiene su propio usuario, su propia contraseña y su propio rol. La máquina mantiene esa lista por sí misma — es su propia lista de personal y no se solapa con nada más.

Los administradores, en particular, no son espectadores. Las sesiones de reproducción de **Sesiones** son las personas que ven los canales, y no tienen nada que ver con el acceso al panel. No hay que mezclar las dos listas: significan cosas distintas, viven tiempos distintos y un error en una cuesta algo completamente distinto que un error en la otra.

3.3.1 El primer inicio de sesión

Una máquina recién instalada aún no conoce administradores — no hay quién los cree. Solo existe una forma de entrar: las credenciales estáticas que el paquete generó durante la instalación. El usuario es `admin`, la contraseña es el valor de `EDIT_AUTH_PASSWORD` en `/etc/mcaster/secrets.env` (el archivo solo lo puede leer root; véase [Instalación](#)).



The image shows a login screen with a dark background. In the center, there is a light blue rectangular area. At the top of this area, the text 'Iniciar sesión' is displayed in a bold, white font. Below this, there are two input fields. The first field is labeled 'Usuario *' in a light gray font. The second field is labeled 'Contraseña *' in a light gray font. At the bottom of the light blue area, there is a large teal button with the text 'Iniciar sesión' in a bold, black font.

El panel acepta esas credenciales y pide de inmediato una cuenta nominativa:

Cree el primer administrador

Ha iniciado sesión con credenciales estáticas. Cree una cuenta nominal para trabajar con su propio usuario.

Al menos 8 caracteres

Crear

[Continuar con credenciales estáticas](#)

La cuenta creada aquí recibe el rol **superadmin** — es el único rol que gestiona a los demás administradores. A partir de ahora trabaje bajo ella y no bajo las credenciales estáticas. La razón es simple: las credenciales estáticas son compartidas y no responden a la pregunta de quién hizo qué.

El enlace **Continuar con credenciales estáticas** omite el paso. La oferta vuelve en el siguiente inicio de sesión — el panel sigue recordándolo mientras la lista de administradores esté vacía.

3.3.2 Las credenciales estáticas son la entrada de emergencia

Las credenciales estáticas no desaparecen cuando ya existen administradores. Es deliberado: permanecen como entrada de emergencia para cuando las cuentas nominativas no ayudan — el único superadmin dejó la empresa, olvidó su contraseña o le desactivaron la cuenta por error. Entrando con ellas, siempre puede crear un nuevo superadmin.

Trátelas como trata la llave de la sala de servidores:

- la contraseña vive en `/etc/mcaster/secrets.env`, y solo root puede leer el archivo;
- se cambia en ese mismo archivo, y la máquina recoge el nuevo valor al reiniciar el servicio;
- a quien necesite acceso cotidiano, créele una cuenta nominativa en lugar de entregarle esta contraseña.

3.3.3 La sección «Administradores»

La sección solo es visible para un superadmin. Para los demás roles no aparece ni en el menú ni es alcanzable por enlace directo — el rechazo viene del servidor, y el menú simplemente deja de ofrecer lo que el rol de todos modos no obtendría.

The screenshot shows the Mcaster web interface. On the left is a sidebar with a menu including 'Panel de control', 'Streamers', 'DVR del clúster', 'Layouter', 'Streams', 'Plantillas', 'Importar desde Flussonic', 'VOD', 'Auth & DRM', 'Sesiones', 'Administradores' (highlighted), and 'Registro de auditoría'. At the bottom of the sidebar are options for 'Tema claro' and 'Cerrar sesión', and a language dropdown set to 'Español'. The main area is titled 'Administradores' and contains a table with the following data:

USUARIO	ROL	ESTADO	ETIQUETAS DE CONTENIDO
anna	superadmin	Activo	Editar Eliminar

A 'Añadir administrador' button is located in the top right corner of the main area.

La lista tiene tres columnas, que responden a tres preguntas:

- **Usuario** — el nombre con el que la persona inicia sesión.
- **Rol** — lo que le está permitido hacer.
- **Estado** — **Activo**, **Contraseña temporal** (la cuenta existe, pero su dueño aún no ha sustituido la contraseña que se le entregó) o **Deshabilitado** (la cuenta está, el acceso no).

3.3.4 Roles

Un administrador tiene exactamente un rol, elegido entre tres:

- **superadmin** — todo, incluida esta sección y la sustitución del documento de licencia.
- **admin** — todo el trabajo cotidiano: streams, plantillas, máquinas del clúster, el archivo, la protección de reproducción. Exactamente dos cosas están cerradas: la sección de administradores y la sustitución de la licencia.
- **viewer** — solo observación: el panel de control y las sesiones. La configuración está cerrada por completo, incluida la lectura.

El viewer merece una nota, porque prohibir la **lectura** de la configuración parece excesivo. No lo es: los documentos de configuración llevan secretos por construcción — contraseñas de fuentes ajenas, claves, credenciales de máquinas. Un documento no se puede entregar con los secretos recortados por el camino, así que a un viewer no se le entrega en absoluto.

El servidor comprueba el rol en cada petición, no solo al iniciar sesión.

3.3.5 Añadir un administrador

El botón **Añadir administrador** abre el formulario:

Añadir administrador

Usuario

oleg

Contraseña

••••••••

Al menos 8 caracteres

Repita la contraseña

••••••••

Rol

admin

▼

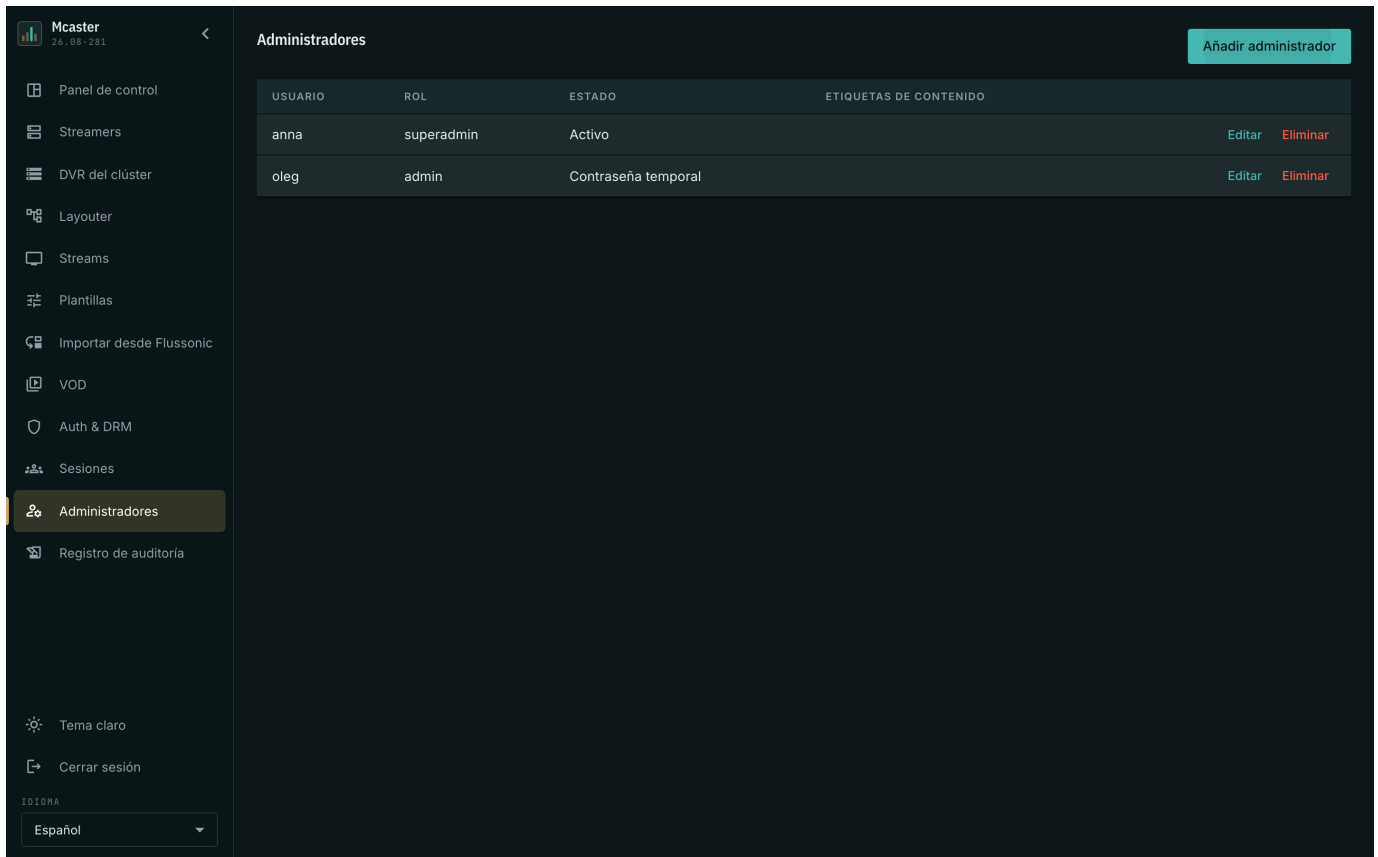
☒ Exigir cambio de contraseña en el primer inicio de sesión

Cancelar

Crear

Introduzca aquí una contraseña temporal y entréguela a la persona como usted entregue las contraseñas. Marque la casilla **Exigir cambio de contraseña en el primer inicio de sesión**: una contraseña que usted inventó por otra persona también la conoce usted, lo cual significa que ya no es un secreto solo de ella.

La cuenta nueva aparece en la lista en el estado **Contraseña temporal**:



The screenshot shows the 'Administradores' (Administrators) section of the Mcaster interface. On the left is a sidebar with a menu including 'Panel de control', 'Streamers', 'DVR del clúster', 'Layouter', 'Streams', 'Plantillas', 'Importar desde Flussonic', 'VOD', 'Auth & DRM', 'Sesiones', 'Administradores' (highlighted), and 'Registro de auditoría'. At the bottom of the sidebar are options for 'Tema claro' and 'Cerrar sesión', along with a language dropdown set to 'Español'. The main content area is titled 'Administradores' and features a teal 'Añadir administrador' button. Below the title is a table with columns: 'USUARIO', 'ROL', 'ESTADO', and 'ETIQUETAS DE CONTENIDO'. The table contains two entries: 'anna' with role 'superadmin' and status 'Activo', and 'oleg' with role 'admin' and status 'Contraseña temporal'. Each entry has 'Editar' and 'Eliminar' links.

USUARIO	ROL	ESTADO	ETIQUETAS DE CONTENIDO
anna	superadmin	Activo	Editar Eliminar
oleg	admin	Contraseña temporal	Editar Eliminar

En su primer inicio de sesión, el dueño de esa cuenta cae no en el panel, sino en el formulario de cambio de contraseña:

Cambie su contraseña

Esta cuenta se creó con una contraseña temporal.
Cámbiela para continuar.

Contraseña actual

Nueva contraseña

Al menos 8 caracteres

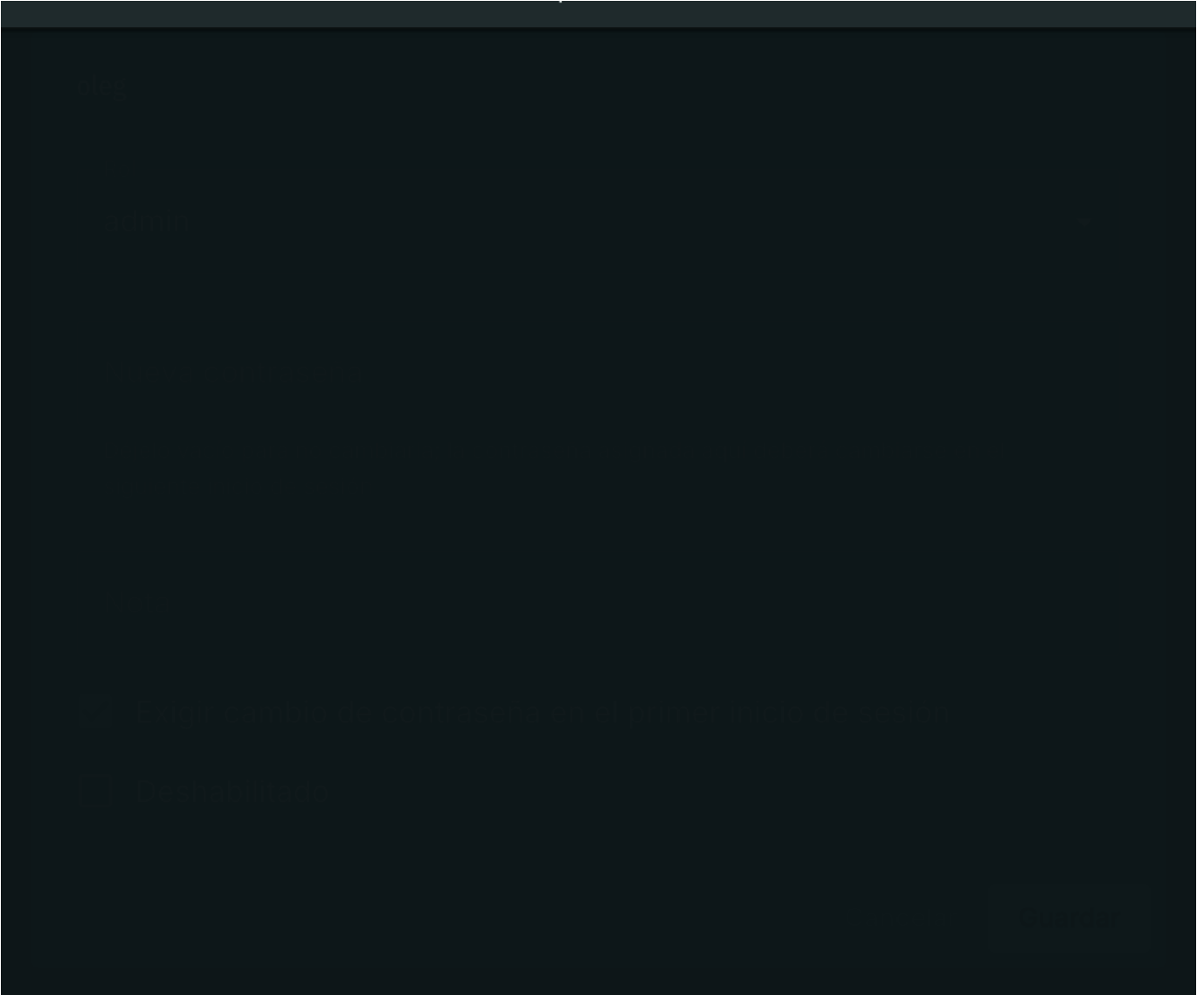
Repita la nueva contraseña

Cambiar contraseña

No se abre nada más hasta que se cambie la contraseña — es el servidor el que rechaza, no una ventana con aspecto severo: conocer las direcciones no sirve para saltarse el formulario. Una vez cambiada, la persona queda en el panel de inmediato, sin un segundo inicio de sesión.

3.3.6 Editar, restablecer contraseña y desactivar el acceso

El botón **Editar** de una fila abre todo lo que se puede cambiar en una cuenta existente:



- **Rol** — el nuevo rol surte efecto de inmediato: las sesiones de la cuenta se cierran y la persona inicia sesión de nuevo con los nuevos derechos.
- **Nueva contraseña** — así se recupera a quien olvidó la suya. El panel no le mostrará a nadie su propia contraseña, por lo que una contraseña olvidada no se recupera, sino que se sustituye. La contraseña emitida aquí también es temporal: su dueño la cambia en el siguiente inicio de sesión. Un campo vacío significa «no tocar la contraseña».
- **Nota** — para qué sirve esta cuenta y de quién es. Empieza a valer la pena cuando ya hay una docena.
- **Deshabilitado** — el acceso se cierra de inmediato, junto con cualquier sesión ya abierta, mientras que la cuenta en sí permanece.

Administradores

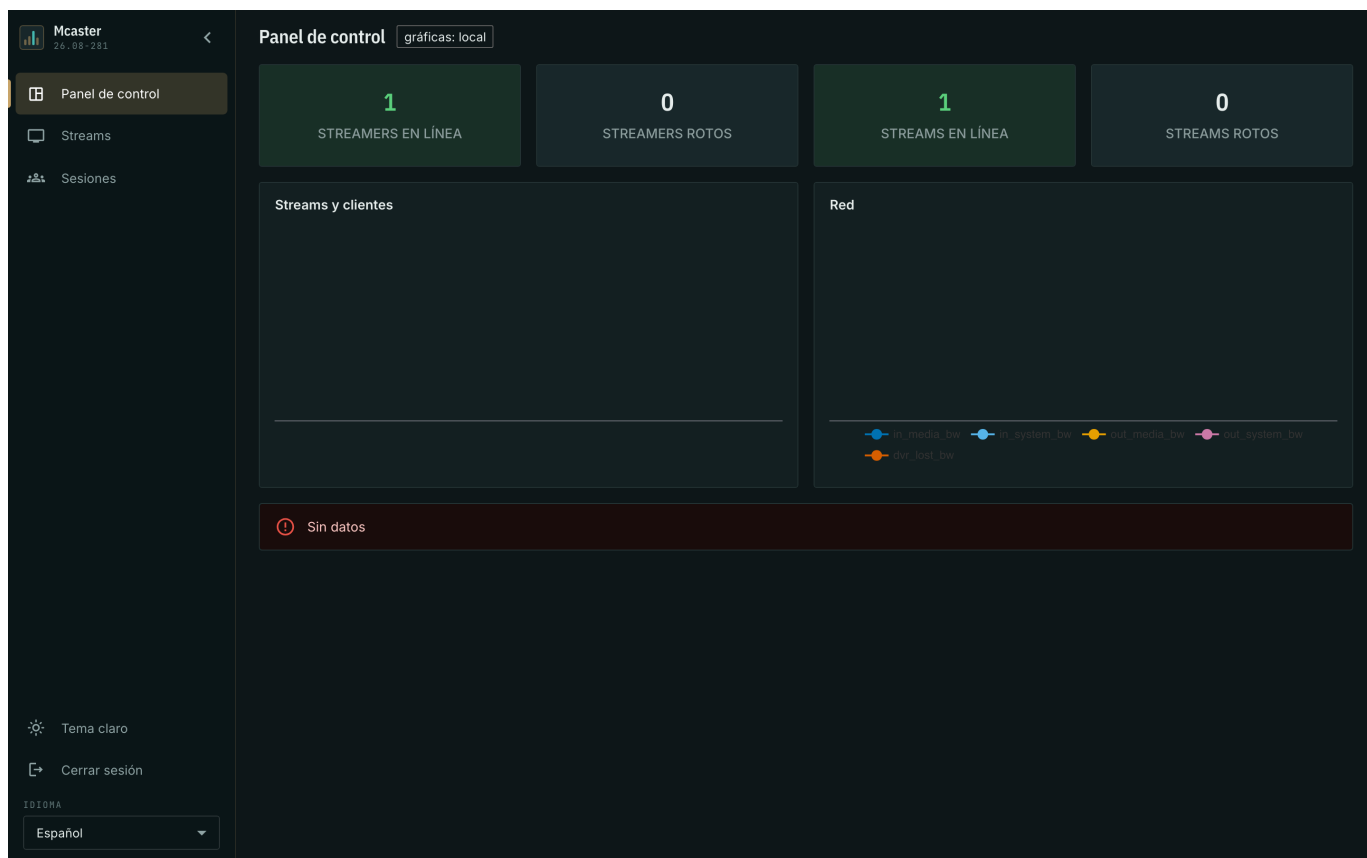
Añadir administrador

USUARIO	ROL	ESTADO	ETIQUETAS DE CONTENIDO
anna	superadmin	Activo	Editar Eliminar
nina	viewer	Activo	Editar Eliminar
oleg	admin	Contraseña temporal	Editar Eliminar

Desactivar es a lo que se recurre la mayor parte del tiempo: alguien se fue de vacaciones, cambió de puesto dentro de la empresa, perdió el portátil. El botón **Eliminar** borra la cuenta para siempre; elimine cuando la cuenta se creó por error.

3.3.7 Qué ve un viewer

Así se ve el mismo panel bajo el rol **viewer**: del menú quedan solo el panel de control y las sesiones, y las secciones de configuración no se muestran.



Es un rol cómodo para el turno de guardia y para cualquiera que necesite ver qué pasa sin cambiar nada.

3.3.8 La sesión en el navegador

El panel no guarda la contraseña en el navegador. Tras el inicio de sesión funciona con un pase emitido por el servidor, que renueva por sí mismo mientras la pestaña está abierta — no le vuelven a pedir la contraseña a lo largo de la jornada laboral.

El botón **Cerrar sesión** cierra la sesión en el servidor, y no solamente la pestaña. Eso importa cuando usted deja la máquina de otra persona: cerrar una pestaña no quita el acceso; cerrar la sesión, sí.

4. Captura de señal

4.1 Captura de multicast y MPTS

Un flujo de transporte por UDP es la entrada principal de la cabecera. La señal llega como un servicio aparte en un grupo (SPTS) o como un multiplex en el que el programa necesario es uno de una docena (MPTS). Mcaster acepta ambos casos, y en el segundo recibe el tronco **una sola vez** para todos sus programas, por muchos streams que usted monte a partir de él.

4.1.1 Conecte el grupo

Cree un stream y pulse **+ Añadir fuente** en la pestaña **Fuentes**. El protocolo es **UDP MPEGTS**:

- **Host** — la dirección del grupo multicast, por ejemplo `239.255.1.77`, o `0.0.0.0` para unicast en un puerto;
- **Puerto** — el puerto del grupo;
- **Tiempo de espera del par (ms)** — cuánto silencio tolerar antes de reiniciar la captura; el campo vacío significa el valor por defecto.

Pulse **Guardar**. Si en el campo **Host** hay un grupo multicast, la estación se suscribe a él por sí misma (IGMP para IPv4, MLD para IPv6). El unicast se diferencia solo en la dirección: `0.0.0.0` significa «escuchar el puerto en todas las interfaces».

4.1.2 Un programa de un tronco MPTS

Un multiplex de un satélite o de un socio lleva varios servicios en un mismo grupo. El necesario se elige con el campo **Programa (PNR)** del mismo origen: cree un stream por servicio, todos con el mismo grupo y puerto y con números de programa distintos.

El campo vacío significa «la primera con contenido» — así lo dice el formulario debajo del campo.

Lo que importa aquí en la práctica:

- **El tronco se recibe una sola vez.** Dos streams en el mismo grupo y puerto comparten el socket y el demultiplexor: la red ve una sola suscripción y la máquina hace un único análisis del transporte. Diez programas de un tronco no cuestan diez capturas.
- **Sin número indicado, se toma el primer programa que realmente lleva medios.** No el primero según el PSI: las tablas de los troncos suelen estar desfasadas y anuncian servicios cuyos PID fueron recortados hace tiempo.
- **Un número ilegible lo rechaza el formulario**, en lugar de interpretarse como «todo el tronco»: de lo contrario el stream acabaría siendo una mezcla de todos los programas del multiplex.
- **Los contadores se atribuyen por suscriptor**, mientras que los errores del tronco en su conjunto se duplican a todos: por stream se ven tanto los PID propios como el problema ocurrido con el transporte entero.

4.1.3 Captura de un remitente concreto (SSM)

Cuando varias fuentes emiten a un grupo y solo interesa una de ellas, ponga su dirección en el campo **Origen (SSM)**.

El kernel se suscribe al grupo indicando el remitente (`IP_ADD_SOURCE_MEMBERSHIP`, IGMPv3) y descarta el tráfico ajeno antes de que llegue a la estación. Soportado para IPv4.

4.1.4 Captura por HTTP

El mismo flujo de transporte servido por HTTP es la forma habitual de tomar la señal de un socio cuando entre ustedes no circula multicast. El protocolo del origen es **HTTP MPEGTS**, con una **URL** y, si hace falta, un **Tiempo de espera del par (ms)**.

La transferencia chunked está soportada. Una conexión rota y una conexión viva pero silenciosa se distinguen y aparecen de forma distinta en el estado de la entrada — son dos diagnósticos diferentes: «nos desconectaron» y «dejaron de enviar».

Una entrada HTTP tiene activado por defecto el búfer de entrada (30 segundos): un origen así casi siempre llega a ráfagas y, sin el búfer, las ráfagas se irían a la salida.

4.1.5 Qué se analiza del stream

Vídeo H264, HEVC, MPEG-2; audio AAC, AC3, EAC3, MPEG-2 audio; teletexto DVB y subtítulos DVB con sus descriptores, subtítulos CEA-608/708, metadatos KLV, marcadores SCTE-35, descriptores de idioma de las pistas del PMT.

Los marcadores SCTE-35 se reciben y sobreviven al tránsito hasta la salida. Procesarlos — sustituir, compensar, insertar — aún no está en Mcaster.

4.1.6 Qué debe cumplirse en la red

- **El multicast no viaja por internet.** La estación debe estar en el mismo segmento que el origen del grupo, o la red entre ellos debe encaminar multicast.
- **Quien se suscribe es la estación.** La petición IGMP sale de ella, y los conmutadores con IGMP snooping entregan el stream solo a los suscriptores. No hay nada que configurar en la estación para ello — lo importante es que la red deje pasar la suscripción.
- **Compruebe desde la máquina de la estación.** El `tcpdump -i <iface> host 239.255.1.77` ejecutado allí responde a la pregunta «¿llegan los paquetes?», y ninguna comprobación desde una máquina vecina lo sustituye.

4.1.7 Aplicación de los cambios

El cambio de grupo, puerto, número de programa o URL reinicia el origen. El cambio de un solo tiempo de espera se aplica en caliente, sin interrumpir la captura.

4.1.8 Qué viene después

- [Añada una entrada de reserva](#) — un segundo origen y las reglas de conmutación.
- [Errores de entrada](#) — qué significan los contadores cuando la señal se degrada.
- [Devuelva la señal a la red.](#)

4.2 Captura por SRT y RIST

SRT y RIST llevan la señal a través de una red que usted no controla: por internet, entre ciudades, desde una unidad móvil. Las pérdidas en una red así son inevitables, y ambos transportes las recuperan por retransmisión mientras quede tiempo para hacerlo.

4.2.1 SRT

En la pestaña **Fuentes** pulse **+ Añadir fuente** y elija el protocolo **SRT**. La estación espera la conexión en un puerto UDP dedicado y el remitente se conecta a él.

Campo	Qué define
Host, Puerto	la dirección y el puerto UDP en los que se espera la conexión
Frase de paso	la clave de cifrado AES; si no coincide, la conexión se rechaza en el handshake
Stream ID	el identificador esperado del remitente
Latencia de recepción (ms)	el presupuesto para recuperar pérdidas (véase más abajo); 120 por defecto
Tiempo de espera del handshake (ms), Tiempo de espera del par (ms)	los tiempos de espera de conexión y de inactividad

Una dirección `srt://` pegada entera se descompone en los campos, y la **URL SRT** reconstruida se muestra al lado — eso es lo que se copia y se entrega al remitente.

4.2.2 La latencia es un presupuesto, no un retraso por retrasar

La **Latencia de recepción** define cuánto tiempo espera un paquete en el búfer de recepción antes de entregarse. Un paquete perdido solo llega a tiempo de volver si la solicitud de retransmisión y la respuesta caben en esa ventana — de ahí la regla práctica escrita en la propia ayuda del campo: **no menos de tres o cuatro RTT** hasta el origen.

En un enlace con RTT de 100 ms, los 120 ms estándar no dejan sitio ni para un solo intento de recuperación: una pérdida se convierte en un agujero en la señal. El precio de subirla es honesto y predecible — exactamente esa demora en la salida y memoria para el búfer de recepción.

El valor se elige a partir del RTT medido y no de la sensación: el RTT y su variación están en los [contadores de la conexión](#).

4.2.3 Qué hace el transporte

Está implementado el mecanismo ARQ completo: confirmaciones (ACK/ACKACK), solicitudes de retransmisión (NAK) con un período ajustado al RTT medido, entrega en orden desde el búfer de latencia, descarte de los paquetes irremediablemente tarde (too-late drop) y keepalive para conexiones inactivas.

La consecuencia práctica para una cabecera: **las pérdidas en el enlace no tienen por qué volverse defectos de la señal**. Mientras el presupuesto alcance, el agujero se cierra con una retransmisión y nunca llega a la salida. Cuando el presupuesto no alcanza, los contadores lo dicen: crecen los descartes por tarde, no solo las retransmisiones.

4.2.4 RIST

El protocolo **RIST** se añade con el mismo **+ Añadir fuente**. Hay dos diferencias respecto a SRT, y ambas se ven en el formulario:

- **Perfil RIST — Simple (par de puertos)**, donde los medios y el RTCP ocupan puertos vecinos (el puerto de medios debe ser par), o **Main (tunel GRE)**, donde por el cable pasa un único **Puerto del tunel**. El tunel además tiene un **Rol en el tunel** — conectarse uno mismo o esperar — y una **Contraseña PSK** que cifra el contenido; debe coincidir en ambos lados.
- **Buffer de recepcion (ms)** — el mismo, en significado, presupuesto de recuperación que la latencia de SRT; 1000 por defecto. Al lado están **Seccion de reordenamiento (ms)**, **Numero de reintentos** y la **Interfaz de multicast** para recibir un grupo.

4.2.5 Aplicación de los cambios

El cambio de dirección, puerto, clave o identificador reinicia el origen. El cambio de los tiempos de espera solos se aplica en caliente.

4.2.6 Qué viene después

- [Envíe la señal más adelante por SRT](#) — a un destino fijo o para que un socio la tome.
- [Añada una entrada de reserva](#) — SRT en pareja con un multicast.
- [Errores de entrada](#) — cómo distinguir un origen muerto de un enlace que se degrada.

4.3 Publicación SRT

La [captura por SRT](#) funciona así: usted le dice de antemano a la estación dónde esperar la señal, y el remitente debe marcar exactamente allí. La publicación invierte el esquema — un corresponsal con OBS o un codificador móvil marca al clúster por su cuenta, con una sola dirección y un solo puerto, sin saber ni elegir de qué máquinas se compone el clúster. La captura arranca en el nodo que atendió la llamada.

Así se conecta a los autores externos: un reportero con un teléfono, el estudio de un socio, una cámara en un evento. Reciben una sola cadena de conexión — del resto se encarga el clúster.

4.3.1 Cómo el clúster atiende la llamada

- El puerto UDP dedicado del stream lo escuchan **todos los nodos del clúster** — marque al que marque, la puerta es la misma.
- El que llama pasa la comprobación de la **frase de paso** en el propio handshake; después el stream queda anclado al nodo que atendió la llamada y arranca allí.
- Una publicación caída espera su regreso: tras unos segundos de silencio la conexión se cierra y el stream vuelve a la espera. Un codificador que vuelve a marcar a otro nodo se lleva el stream consigo — el anclaje sigue a la llamada.
- Mientras la publicación está en marcha, la puerta está ocupada: un segundo codificador que marque al mismo puerto recibe un rechazo de «ocupado» y no puede tumbar la transmisión en curso.

Las fuentes son opcionales para tal stream: puede vivir solo de la publicación y arrancar con la primera llamada. Una fuente configurada junto a la publicación actúa de reserva — el directo la supera.

4.3.2 Una dirección para todos los codificadores

El codificador necesita una sola dirección, y el clúster tiene varios nodos. El ajuste **Dirección de publicación** (menú **Ajustes de central**) es el nombre o la IP que la consola pone en la cadena de conexión. Lo que hay detrás de esa dirección — un nombre DNS con varios registros A, una VIP o un balanceador UDP — es decisión de la instalación; esta página cubre la variante DNS como la que no requiere software adicional. El ajuste vacío significa el host de la consola y solo sirve para una máquina única.

Publicación SRT

Un nombre público o IP al que se conectan los codificadores. Lo que hay detrás (DNS con varios registros A, VIP, balanceador UDP) depende de la instalación. Vacío — el host de la consola (caja única).

Dirección de publicación

Direcciones visibles de los nodos — candidatas a registros DNS del nombre:

192.168.28.240 — node1
192.168.28.242 — node3
192.168.28.241 — node2

4.3.3 Escenario: un nombre DNS con varios registros A

En SRT no hay redirecciones: el cliente marca por UDP a una dirección concreta, y el protocolo no sabe decir «pregunte en el nodo de al lado». Lo que sí hacen los clientes SRT — OBS, Larix, VLC — es recorrer los registros A: cuando una dirección no responde, prueban los registros del nombre por su cuenta, gastando unos tres segundos por cada uno muerto. Un nombre con un registro A por nodo da conmutación por error a cargo del propio cliente: un nodo muere — el codificador marca al siguiente.

El orden de configuración:

1. **Declare las direcciones públicas de los nodos.** En la ficha de cada nodo rellene la **Dirección pública de medios** — el campo ofrece elegir entre las direcciones que el nodo ve en sus interfaces; detrás de NAT la correcta la sabe solo usted, escríbala a mano.
2. **Consulte la pista.** Bajo el ajuste **Dirección de publicación** la consola lista los candidatos a registros DNS — una dirección por nodo vivo, con el nombre del nodo. Esa es exactamente la lista de lo que va a la zona.
3. **Cree los registros A.** En su zona DNS cree un nombre — por ejemplo, `publish.example.com` — con un registro A hacia la dirección de cada nodo de la lista.
4. **Ponga el nombre en «Dirección de publicación»** y guarde. El clúster resuelve el nombre periódicamente y coteja el resultado con el registro de nodos: junto al ajuste se ve que el nombre lleva a nodos vivos. Un nombre que no se resuelve, o que lleva solo a nodos apagados, enciende una advertencia — el error clásico de mantenimiento «nodo retirado, registro olvidado» se ve de inmediato, no en la primera transmisión rota. Desde ese momento las URL del codificador de todos los streams se arman con el nombre.

Dos salvedades honestas:

- Varios registros A son **conmutación por error, no balanceo de carga**: los resolutores cachean y barajan los registros a su manera, y los codificadores de una misma red pueden caer todos en el mismo nodo. Un sesgo de anclajes entre nodos es un cuadro normal.
- **No todo cliente móvil recorre los registros.** Algunas aplicaciones toman la primera dirección y siguen insistiendo en ella. Para esas es más seguro configurar dos direcciones en el perfil del codificador — principal y de reserva.

4.3.4 Configuración del stream

Abra el stream en la pestaña **Streams** y busque la sección **Publicación SRT**:

Campo	Qué define
Puerto UDP	el puerto de publicación; un puerto — un stream, en todos los nodos a la vez
Frase de paso	la clave de cifrado AES, de 10 a 79 bytes; si no coincide, la llamada se rechaza en el handshake

La frase de paso es opcional, pero un puerto sin ella es una puerta abierta: podrá publicar cualquiera que marque. Déjela vacía solo en una red de confianza — el formulario lo advierte.

Publicación SRT

Ingesta en un puerto UDP dedicado: el codificador recibe una sola línea — dirección y passphrase. Las fuentes son opcionales; una fuente junto a la publicación actúa como respaldo.

Puerto UDP
9003

Passphrase

Quitar la publicación

Opcional, 10-79 bytes; se recomienda ASCII

 El puerto está abierto sin passphrase: sin cifrado y sin candado, cualquiera que se conecte puede publicar. Solo para redes de confianza.

SRT
Larix
Moblin



URL del codificador
srt://publish.lan:9003

Debajo de los campos se arma la **URL del codificador** lista — `srt://dirección:puerto?passphrase=...` — con botón de copiar y código QR: la dirección con la clave entran en el codificador móvil por la cámara, no por el teclado. La dirección de la cadena es la misma **Dirección de publicación** de la sección anterior: configúrela antes de repartir la cadena, o el codificador se llevará la dirección de un nodo concreto.

Junto al QR están las pestañas **Larix** y **Moblin**: la misma puerta como enlace de importación listo para esas aplicaciones — se escanea con su cámara y la conexión aparece sola en sus ajustes.

Publicación SRT

Ingesta en un puerto UDP dedicado: el codificador recibe una sola línea — dirección y passphrase. Las fuentes son opcionales; una fuente junto a la publicación actúa como respaldo.

Puerto UDP
9001

Passphrase
korovamoloko-live

Quitar la publicación

Opcional, 10-79 bytes; se recomienda ASCII

SRT

Larix

Moblin



URL del codificador
srt://publish.lan:9001?passphrase=korovamoloko-live

4.3.5 Conexión del codificador

En OBS: **Configuración** → **Emisión** → **Servicio: Personalizado...**, y la cadena entera de la ficha del stream en el campo **Servidor**:

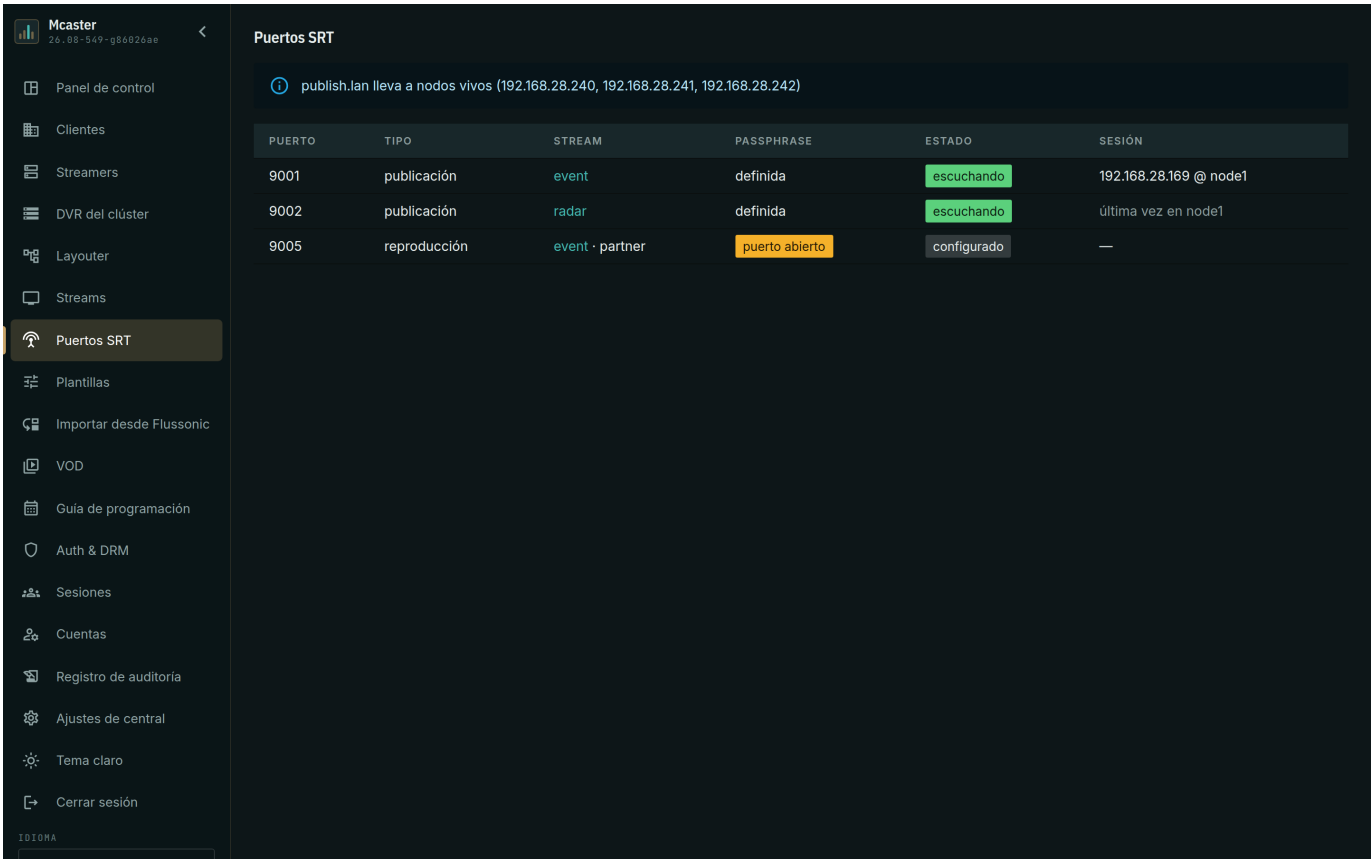
```
srt://publish.example.com:9000?passphrase=su-clave
```

Al codificador móvil muéstrole el QR de la misma sección. Tras arrancar la emisión, el stream se levanta en un par de segundos — se ve en la lista de streams y en la página de puertos.

4.3.6 El resumen de puertos SRT

La página **Puertos SRT** (en el menú, bajo **Streams**) es el mapa de todas las puertas SRT del clúster: puertos de publicación y de **entrega**, de qué stream es cada uno, si tiene frase de paso, si de verdad está escuchando. También muestra:

- **la elección de puerto para un stream nuevo** — los números ocupados están a la vista, un puerto duplicado queda marcado;
- **una publicación en vivo** — la dirección del codificador y el nodo que atendió la llamada;
- **el anclaje entre llamadas** — para un stream cuyo codificador se cayó y aún no ha vuelto, el nodo donde el stream espera: una llamada a él se reanuda sin consultar a central, una llamada a otro nodo se lleva el stream allí;
- **la comprobación de la dirección de publicación** — la misma línea de estado que junto al ajuste.



4.3.7 Cuándo se rechaza una llamada

Qué pasó	Qué ve el codificador
La frase de paso no coincidió	un rechazo en el handshake, sin conexión
El puerto está ocupado por una publicación en curso	un rechazo de «ocupado»; la transmisión en marcha no se interrumpe
La publicación se quedó en silencio	tras unos segundos de silencio la conexión se cierra y el stream espera la próxima llamada

4.3.8 Qué sigue

- [Captura por SRT y RIST](#) — el esquema clásico cuando la dirección de la fuente se conoce de antemano.
- [Entrega por SRT](#) — pasar la señal recibida más allá.
- [Monitoreo](#) — contadores de conexión y calidad de la captura.

4.4 Recepción de ST 2110

ST 2110 es la señal de estudio transportada por una red IP: el vídeo sin comprimir, el audio y los datos auxiliares viajan en **grupos multicast distintos**, atados por un reloj PTP común. McaSTER toma esa señal como una fuente de stream corriente: las pistas entran en la cadena, se transcodifican, se graban y se entregan con los mismos medios que una señal de satélite o una fuente SRT.

Una cosa distingue al 2110 de cualquier otra entrada, y la configuración empieza por ella: **aquí la fuente no es una dirección sino un conjunto de pistas**. Vídeo, audio y datos auxiliares son tres grupos distintos con tres juegos de parámetros distintos, y el stream se arma con ellos y no con una sola URL.

4.4.1 Qué debe cumplir la red

Un flujo sin comprimir son gigabits por segundo, y la red se prepara para ello antes de que nadie abra el panel.

- **Ancho de banda por flujo.** 1080p60 a 10 bits son unos 2,5 Gbit/s y 260 mil paquetes por segundo. Una interfaz de 10 Gbit/s aguanta tres flujos así, no cuatro: el resto es margen para las ráfagas.
- **El puerto lo elige el operador, no la estación.** La suscripción sale por la interfaz indicada en los ajustes de la entrada, y ahí es donde el switch entrega el grupo. Un puerto indicado sin ancho de banda suficiente es el diagnóstico más frecuente: el flujo «llega» mientras la imagen se deshace, porque quien descarta paquetes es el switch — y los contadores del propio equipo siguen limpios.
- **PTP es obligatorio.** El reloj de 2110 viene del dominio PTP, y la estación concilia su propia escala de tiempo con él. Un reloj desviado se ve en el estado del stream aparte de los errores de recepción: no son lo mismo.
- **Multicast e IGMPv3.** Los grupos se entregan por la suscripción de la estación; cuando varios emisores alimentan un grupo y usted necesita uno de ellos, su dirección va al campo **Origen (SSM)**.

4.4.2 Conecte las pistas

Cree un stream, abra la pestaña **Fuentes** y pulse **+ Añadir fuente**. El protocolo es **ST 2110**.

Protocolo
ST 2110

Interfaz
dpdk0

Direccionamiento
Controlador (IS-05)

Nombre (dpdk0) o su dirección; no existe el valor «cualquiera»

El controlador trae las direcciones por IS-05. A mano es lo mismo hecho por usted: la dirección y el formato de cada pista los indica usted, para una red sin controlador y para pruebas.

☐ Reducir 10 bits a 8 en la recepción

Ahorra ancho de banda hasta el transcodificador a costa de los dos bits bajos; desactivado por omisión.

Receptores

Video, audio y datos auxiliares los anuncia la propia entrada: el controlador ve tres receptores y trae las direcciones.

Vídeo

☒ Recibir

Conmutado por el controlador: 5e647b64-f5f8-56a8-b9ba-9eb4adccae11

Ajuste de esta pista

Audio

☒ Recibir

Conmutado por el controlador: 9f746893-d917-5944-8f80-bfba7292e210

Ajuste de esta pista

Datos auxiliares

☒ Recibir

Conmutado por el controlador: 9784a111-18f4-5b59-a402-0a44f8c58c69

Ajuste de esta pista

Los campos de toda la entrada:

- **Interfaz** — el nombre del puerto (`dpdk0`) o su dirección. Aquí no existe el valor «cualquiera», y es deliberado: la interfaz decide por dónde entrega el switch el grupo.
- **Direccionamiento** — quién da las direcciones a las pistas. De eso trata la sección siguiente.
- **Reducir 10 bits a 8 en la recepción** — ahorra ancho de banda hasta el transcodificador a costa de los dos bits bajos. Desactivado por omisión.

Los **receptores** los anuncia la propia entrada: vídeo, audio y datos auxiliares se muestran siempre, y no se pregunta por su rol. El controlador ve exactamente esos tres receptores y les trae las direcciones. Una pista así no está en la configuración mientras no haya nada que decir de ella.

Puede haber más pistas de un mismo tipo: una segunda pista de audio para otro idioma se crea con **Añadir pista** en el bloque **Pistas añadidas**. Allí el rol (`audio 2`) lo pone usted, y hay que nombrar el tipo explícitamente — de una cadena libre no se deduce. La edición del rol se aplica cuando el campo pierde el foco.

Cada pista tiene la casilla **Recibir**: desmarcarla mantiene la pista en la configuración con todos sus parámetros, pero no se abre ningún socket.

No olvide **Guardar**: la edición de la entrada llega al servidor solo con el botón.

4.4.3 Dos formas de direccionar una pista

La forma se elige en el campo **Direccionamiento** y vale para toda la entrada, no para una pista suelta: una red o la gobierna un controlador o no, y una pista conmutada por el controlador junto a otra escrita a mano no es un escenario sino un malentendido que alguien tendría que depurar.

- **Controlador (IS-05)** — el camino principal. El operador enciende la entrada y nombra la interfaz; de dónde recibir lo dice el controlador, y para todas las pistas a la vez.
- **A mano** — lo mismo hecho por usted: el operador escribe la dirección y el formato de cada pista. Sirve para una red sin controlador y para pruebas.

El archivo de transporte no está en esa lista, y es deliberado. El SDP lo envía el controlador, y describe una conexión ya hecha en lugar de ser una manera de establecerla: nadie escribe SDP a mano. Una pista conmutada lleva la línea **Conmutado por el controlador** con el identificador del emisor, y debajo el **Archivo de transporte**, solo de lectura. Eso es diagnóstico — la respuesta a «de dónde llega esto ahora mismo» — y no un campo de entrada.

El direccionamiento a mano se ve así:

Interfaz
dpdk0

Direccionamiento
A mano

Nombre (dpdk0) o su dirección; no existe el valor «cualquiera»

El controlador trae las direcciones por IS-05. A mano es lo mismo hecho por usted: la dirección y el formato de cada pista los indica usted, para una red sin controlador y para pruebas.

☐ Reducir 10 bits a 8 en la recepción

Ahorra ancho de banda hasta el transcodificador a costa de los dos bits bajos; desactivado por omisión.

Receptores

Vídeo, audio y datos auxiliares los anuncia la propia entrada: el controlador ve tres receptores y trae las direcciones.

Vídeo
☒ Recibir

Grupo multicast
239.255.5.190

Puerto
16394

Origen (SSM)
10.771.90

Dirección del grupo al que se suscribe la pista Puerto RTP anunciado por el emisor Vacío acepta el grupo de cualquiera

Qué hay en el flujo. Una dirección sin formato se rechaza: el medio sin comprimir no lleva su propia geometría. La frecuencia de cuadro es una fracción como en SDP: 60000/1001, no 59.94.

Ancho
1920

Alto
1080

Profundidad de bits
10

Submuestreo
YCbCr 4:2:2

Frecuencia de cuadros
60

☐ Entrelazado ☐ Transporte por campos (segmented)

Ajuste de esta pista

Una dirección sin formato se rechaza, y no es puntilliosidad: el medio sin comprimir no lleva geometría propia dentro del flujo. Un paquete 2110-20 no declara ni el tamaño del cuadro, ni la frecuencia, ni el submuestreo — el receptor tiene que saberlos de antemano, o no hay contra qué analizar las líneas. De ahí el juego de campos: **Ancho, Alto, Profundidad de bits, Submuestreo, Frecuencia de cuadros, Entrelazado, Transporte por campos (segmented)**.

La frecuencia se escribe como fracción, igual que en SDP: $\frac{60000}{1001}$, no 59.94. El redondeo no es un detalle aquí — con él no se puede reconstruir la rejilla de marcas de tiempo.

De los esquemas de submuestreo, el que hoy llega a la cadena es **YCbCr 4:2:2** con 8 o 10 bits. Los demás — 4:2:0, 4:4:4, RGB — se analizan y se reparten en grupos de píxeles, pero de ellos no se arma ningún cuadro: la pista se detiene y el motivo nombra el esquema y la profundidad.

Los dos campos de un cuadro entrelazado los une la propia estación; el desentrelazado sigue siendo trabajo del transcodificador.

4.4.4 Conmutación por un controlador NMOS

El direccionamiento a mano es el camino de reserva. La estación hace lo que la infraestructura de estudio espera de ella: se anuncia a un controlador por **IS-04** y le entrega la conmutación de sus receptores por **IS-05**. El controlador escribe él mismo el archivo de transporte en la pista — la misma edición de configuración que habría hecho un operador.

El anuncio se configura en los ajustes del nodo, sección **NMOS**:

NMOS

Cómo este nodo se anuncia a un controlador IS-04. Sin esta sección el nodo no aparece en el registro, y las direcciones de los receptores 2110 se indican a mano en el propio stream.

Interfaz dppk0 Puerto anunciado; vacío significa NMOS desactivado	Identidad De ella se derivan los identificadores; vacío significa el nombre del host	Etiqueta Mcaster head-end Se muestra en el controlador; vacío significa el nombre del host
---	--	--

Dónde sirve este nodo las interfaces NMOS

☒ Puerto e interfaz propios (recomendado)
 ☐ Puerto HTTP compartido del nodo

Interfaz de las interfaces NMOS Red de control; vacío significa la interfaz de medios	Puerto de las interfaces NMOS Por defecto 8090
---	--

La activación IS-05 la sirve quien es propietario de la configuración: el clúster para un nodo gestionado, la propia caja si es autónoma. Escucha en el mismo puerto que el resto, así que aquí no hay nada que configurar.

- **Interfaz** — el puerto anunciado. Un campo vacío significa que NMOS está apagado y el nodo no aparece siquiera en el registro.
- **Identidad** — de ella se derivan los identificadores de los recursos; vacío significa el nombre del host. Si la cambia, cambian todos los identificadores del nodo y el controlador ve un dispositivo nuevo.
- **Etiqueta** — lo que se muestra en el controlador.
- **Dónde sirve este nodo las interfaces NMOS** — **Puerto e interfaz propios** (recomendado) o **Puerto HTTP compartido del nodo**. El propio se define con **Interfaz de las interfaces NMOS** (la red de control; vacío significa la interfaz de medios) y **Puerto de las interfaces NMOS** (8090 por omisión).

La elección es sobre el acceso, no sobre la comodidad: la activación IS-05 no se autentica, y en el puerto compartido las interfaces aparecen en todos los puertos HTTP del nodo, incluidos los públicos. Solo la red puede limitar quién llega hasta ellas — por eso se recomienda un puerto aparte en la red de control.

La activación la sirve quien es propietario de la configuración: para un nodo gestionado por un clúster, el propio clúster (en sus ajustes es la sección **NMOS (ST 2110)** con la casilla **Conmutar entradas a través de central**); para una caja autónoma, ella misma. En el lado del nodo no hay nada que configurar para eso.

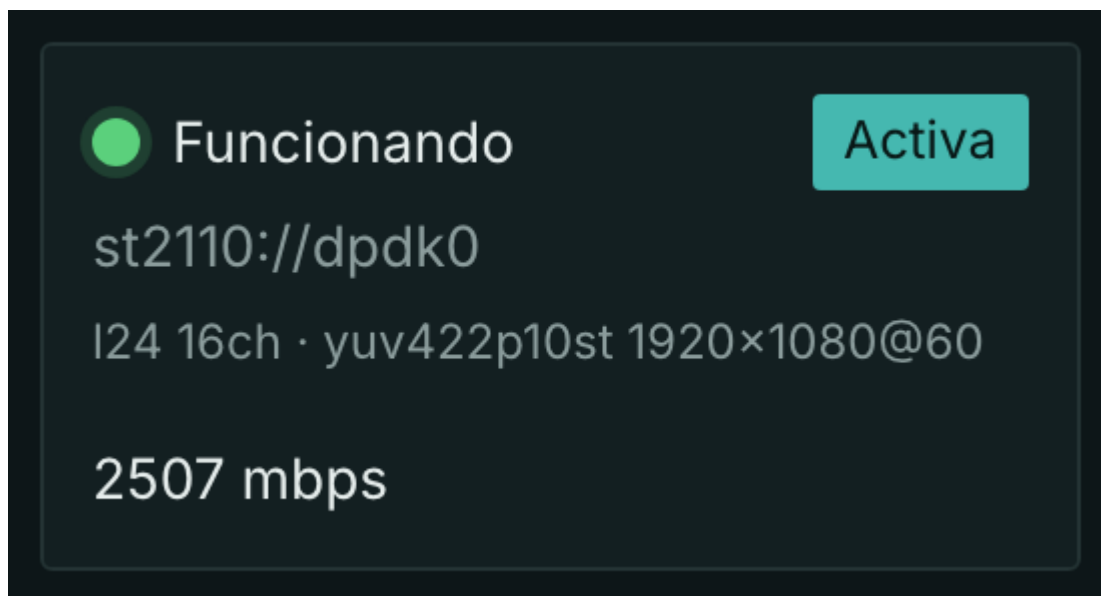
Sobre los receptores importa una cosa: **un nodo nunca tiene receptores de sobra**. El controlador ve exactamente los que se derivan de los streams, y una activación no crea ningún stream. Por eso el stream que se va a conmutar se crea de antemano, con una entrada ST 2110 en modo **Controlador (IS-05)**, donde las pistas aún no tienen direcciones. Mientras el stream no exista, el controlador no tiene nada que conmutar.

Una pista conmutada lo dice en el formulario: **Conmutado por el controlador** con el identificador del emisor. Esa línea la rellena el controlador y no se puede editar.

La estación busca por sí misma un registro y, al encontrarlo, se registra en él y mantiene viva la inscripción con un latido. Si no hay registro en la red, habla directamente con el controlador, punto a punto.

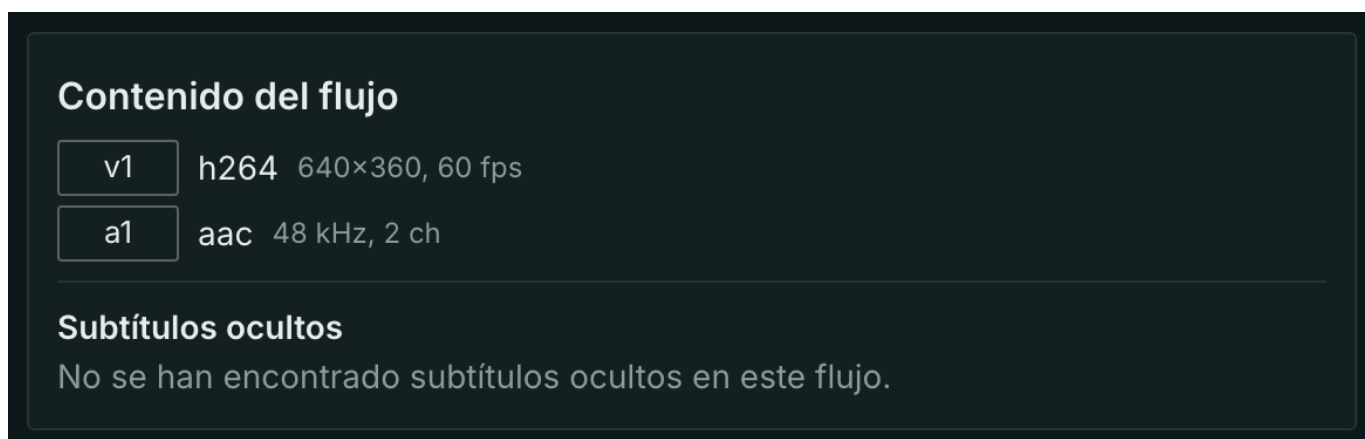
4.4.5 Qué llega al stream

El stream arranca en cuanto empieza a llegar la primera pista y no espera a las demás. El estado de la recepción está en la pestaña **Fuentes**, en el bloque **Salud de la fuente**:



La tarjeta muestra el estado, la dirección de entrada, el conjunto de pistas y el bitrate de recepción. Las cifras de la captura son reales: 2501 Mbit/s es una pista de vídeo 1080p60 de 10 bits más dieciséis canales de audio L24.

El conjunto de pistas está en la pestaña **Configuración básica**, en el bloque **Contenido del flujo**:



El códec nombra la disposición en el cable y no aquello en lo que el flujo se convertirá: `yuv422p10st` para el vídeo de 10 bits y `124` para el audio. La estación no mezcla canales a la baja — toma tantos como declare el emisor y los entrega como una sola pista.

4.4.6 Un flujo sin comprimir necesita transcodificación

Un stream que no tiene más que una entrada 2110 es un estado de funcionamiento, no una avería: arranca, recibe datos y pasa cuadros a los suscriptores. Pero en él no hay segmentación, y **la lista maestra llega sin una sola pista**: una pista sin comprimir no puede ir en ella — un reproductor se rompería analizando su códec en lugar de recibir un honesto «no hay nada que reproducir».

Por eso el panel ofrece un enlace junto al conjunto de pistas — **Transcodificar para reproducir en la web**: abre la pestaña **Transcodificador** con un perfil listo. Después de eso el stream se reproduce, se graba en el archivo y alimenta los envíos como cualquier otro.

Los indicadores de cadena sin comprimir y de segmentación inactiva se muestran aparte en el estado del stream, para que ese estado no se confunda con un fallo.

Las miniaturas de un flujo sin comprimir funcionan sin transcodificador: se toman directamente de la pista sin comprimir, también hacia el archivo.

4.4.7 Datos auxiliares y subtítulos

Una pista del tipo **datos auxiliares** (2110-40) se recibe y se analiza en pares DID/SDID. Lo que ocurre después depende de qué sean esos datos:

- **El teletexto OP-47** llega al stream como una pista de teletexto — la misma que produce el teletexto de MPEG-TS.
- **Los subtítulos CEA-708/608** se analizan y el contador de su par DID/SDID crece, pero en el stream no aparece ninguna pista de subtítulos: esos subtítulos viven dentro del vídeo codificado, y entregarlos a la salida todavía no forma parte de la estación.
- **Los pares desconocidos** se cuentan y se descartan: no existe una pista de «datos auxiliares en bruto» que salga hacia fuera.

Un paquete que no supera su suma de control se descarta con un contador; una paridad rota se cuenta pero no descarta el paquete.

4.4.8 Ajuste fino

Normalmente no hace falta: un campo vacío significa el valor por omisión de la estación, no «apagado». Vive en dos bloques plegados.

Ajuste de esta pista — lo que pertenece a un grupo:

- **Núcleo de CPU** — fija el hilo de recepción de la pista a un núcleo. Cada pista tiene su propio hilo, por eso cada una tiene su propio campo. Vacío lo deja al planificador.
- **Cola AF_XDP** — la cola de la tarjeta que la pista toma para sí, saltándose el socket corriente. Vacío significa socket corriente.

Ajuste de la entrada — lo que es común:

- **Búfer del socket (bytes)** — cuánto retiene el núcleo en el búfer de recepción de cada socket.
- **Cola de cuadros (bytes)** — el límite de la cola de una pista entre la recepción y la cadena. Al desbordarse, lo que sobra se descarta con un contador: la recepción no se detiene.
- **Cuadro de audio (ms)** — la granularidad de los mensajes de audio. Menos significa menos retardo y más mensajes.

4.4.9 Qué está soportado

Capacidad	Estado
Vídeo 2110-20: YCbCr 4:2:2 con profundidad de 8 y 10 bits	sí
Los 10 bits se conservan en la disposición del cable, sin reducción	sí
Otros esquemas 2110-20: 4:2:0, 4:4:4, RGB	no, se analizan pero no dan cuadro — el motivo nombra el esquema
Entrelazado: los dos campos se unen en un cuadro	sí
Transporte por campos (segmented)	la casilla se acepta pero todavía no afecta a la recepción
Audio 2110-30: L16 y L24, cualquier número de canales, <i>ptime</i> del SDP	sí
Datos auxiliares 2110-40: despaquetizado, suma de control, contadores	sí
Teletexto OP-47 como pista de teletexto	sí
Subtítulos CEA-708/608	se analizan y se cuentan, no se entregan como pista
Direccionamiento por archivo de transporte (SDP) enviado por un controlador	sí
Suscripción a un emisor concreto (SSM, IGMPv3)	sí
Varias pistas de un mismo tipo en una entrada	sí
NMOS IS-04: anuncio del nodo, mDNS, registro en un registry	sí
NMOS IS-05: receptores conmutados por un controlador	sí
Que la estación descubra por su cuenta emisores ajenos	no, la dirección la trae un controlador o un operador
SDP con varias líneas de medio en una pista	no, un bloque de direccionamiento describe una pista
Salida 2110 (la estación como emisor)	no

4.4.10 Aplicación de los cambios

Cambiar el grupo, el puerto, el emisor o el archivo de transporte recrea el socket con una suscripción nueva — la recepción de esa pista se interrumpe mientras las demás siguen. Los campos que no afectan a la recepción se aplican al vuelo.

4.4.11 Comprobación

Compruebe en este orden, porque cada paso solo tiene sentido cuando el anterior ha pasado:

- **Los paquetes llegan** — `tcpdump -i <interfaz> host <grupo>` en la propia máquina de la estación. Una comprobación desde una máquina vecina no responde a esta pregunta: la entrega del grupo sigue a la suscripción, y quien la hizo es la estación.
- **Las pistas se levantaron** — la tarjeta del bloque **Salud de la fuente** muestra el conjunto de pistas y el bitrate de recepción.
- **El reloj no se ha desviado** — la desviación del reloj en el estado del stream. Sin sincronización dejan de funcionar el archivo y el rebobinado, y la cura está en el dominio PTP, no en los ajustes de la entrada.
- **La imagen es la esperada** — la vista previa del stream: la geometría y el submuestreo se ponen a mano en el direccionamiento manual, y un error ahí da líneas torcidas en vez de una negativa a recibir.

4.4.12 Un script para un banco sin controlador

Un banco de pruebas normalmente no tiene controlador NMOS, y sin embargo hacen falta las direcciones de los grupos: en la mayoría de los emisores cambian de una activación a otra, así que anotarlas una vez no sirve.

Para ese caso se adjunta un pequeño script — [nmosctl.py](#). Solo necesita Python 3; no tiene dependencias, y para encontrar nodos se usa `avahi-browse`.

```
python3 nmosctl.py discover           # nodos de la red (mDNS)
python3 nmosctl.py list 10.0.0.90:8090 # dispositivos, emisores, receptores
python3 nmosctl.py enable 10.0.0.90:8090 <sender> # habilitar un emisor
python3 nmosctl.py sdp 10.0.0.90:8090 <sender>    # su archivo de transporte
```

El texto que devuelve `sdp` es el que trae un controlador cuando conmuta una pista: en el banco sirve para comprobar el direccionamiento a mano, de donde se copian el grupo, el puerto y el formato.

Los dos tropiezos más frecuentes:

- **Mientras un emisor no esté habilitado no hay archivo de transporte** — `sdp` responde con un error, porque la dirección del grupo aparece solo al activarse.
- **La dirección cambia en cada activación.** Tómela de nuevo en lugar de sacarla de sus apuntes: una suscripción al grupo anterior da cero paquetes en silencio.

4.4.13 Qué sigue

- [Transcodifique el flujo](#) — el perfil tras el cual una señal sin comprimir se reproduce y se graba.
- [Añada una entrada de reserva](#) — una segunda fuente y las reglas de conmutación.
- [Errores de entrada](#) — qué significan los contadores cuando una señal se degrada.
- [Devuelva la señal a la red.](#)

4.5 Reserva de entradas

Una cabecera debe permanecer en el aire cuando muere una fuente. Para eso el stream recibe varias entradas: la estación lleva el aire con la viva de mayor prioridad, vigila las demás y vuelve a la principal cuando esta reaparece.

El par típico en una estación es la recepción satelital como entrada principal y el multicast de un socio como reserva; o dos grupos de dos cabeceras.

4.5.1 Configure varias entradas

En la pestaña **Fuentes** añada una segunda entrada con **+ Añadir fuente**.

- **El orden de la lista es la prioridad:** cuanto más arriba está una entrada, más importante es; el orden se cambia con las flechas **Subir** y **Bajar**. El campo **Prioridad** en la configuración de la entrada prevalece sobre el orden de la lista — el número menor es más importante.
- **Fije el «Intervalo de recomprobación de reserva (s)»** en el bloque **Tiempos de espera de entrada**. Mientras una reserva no está al aire, la estación la sondea periódicamente, y en la lista aparece como **Lista** — comprobada, pero no activa. Sin recomprobación, la salud de la reserva sigue siendo cuestión de fe hasta el fallo mismo, que es justo el momento en que averiguarlo ya es tarde; 0 la desactiva.

Los estados de la entrada en la lista son **Funcionando**, **Lista**, **Iniciando**, **Degradada**, **Muerta** y **Sin comprobar**.

4.5.2 Reglas de muerte y degradación

Cada entrada lleva una placa **Opciones de la fuente**; los campos vacíos heredan las reglas comunes del stream del bloque **Tiempos de espera de entrada**. Así los umbrales se fijan una vez por stream y se anulan uno a uno, mientras que la placa plegada muestra con chips qué es exactamente lo configurado.

Los campos se agrupan de a tres, según lo que hace la estación cuando se rompe una regla.

- **Cuando la fuente está muerta** — fallos duros, conmutación inmediata: **Tiempo de espera de fuente (s)** (no hay ni un fotograma), **Tiempo de espera de vídeo** (no hay vídeo), **Bitrate máximo** (bitrate por encima del techo — protección contra basura en la entrada).
- **Cuando la fuente está degradada** — fallos blandos, conmutación solo si existe una fuente mejor: **Tiempo de espera de audio** (el audio se fue mientras el vídeo vive), **Bitrate mínimo (kbit)** (bitrate por debajo del suelo), **Límite de errores CC** (violaciones de continuidad MPEG-TS dentro de la ventana limpia; el campo existe solo para protocolos de transporte — UDP, HTTP MPEGTS, SRT, RIST).
- **Recuperación y reintentos** — **Ventana limpia (s)**, **Tiempo máximo de reintento** y **Prioridad**.

La degradación va de la calidad de la señal, y dejar una entrada degradada por una muerta no tiene sentido. Por eso una regla blanda no tira el aire, solo baja la entrada en la cola. El motivo de una marca la estación lo nombra directamente: «errores CC por encima del límite», «bitrate por debajo del suelo», «sin audio», «sin cuadros».

4.5.3 La ventana limpia

La **ventana limpia** es la duración de un tramo sin una sola violación. Una entrada cuenta como operativa si y solo si no ha roto ninguna regla durante esos segundos, y la misma ventana se le exige a una entrada revivida antes de volver al aire.

Es la protección contra el péndulo: una fuente que se levanta diez segundos y vuelve a caer ya no sacude el aire. La contrapartida es honesta y predecible — el retorno a la principal siempre queda pospuesto por la duración de la ventana.

4.5.4 Qué ocurre en el momento de la conmutación

- **Una entrada se vuelve activa solo cuando está lista.** Debe entregar los parámetros del stream y el primer fotograma clave; hasta entonces la actual sigue en el aire. Justo por eso la conmutación no da pantalla negra.
- **El empalme no es perfecto.** El fotograma clave de la entrada nueva es el punto de empalme, y el receptor lo verá: un fotograma puede repetirse o perderse. No espere de este mecanismo una conmutación hitless de nivel ST 2022-7.
- **La divergencia de parámetros se ve con antelación.** La estación compara los parámetros de las entradas y muestra la divergencia antes del fallo — una reserva con otra resolución u otro conjunto de pistas se nota pronto, y no en el momento de la conmutación.

- **Una entrada caída se reconecta con una pausa creciente**, y cuando se agotan todas las entradas, la rotación vuelve a empezar desde la primera.

4.5.5 La cortinilla

El bloque **Cortinilla cuando caen todas las fuentes** le da al stream una última línea de defensa: en el campo **Ruta del archivo en el streamer** se pone la ruta a un archivo preparado, y el **Tiempo de espera (s)** decide a cuántos segundos de la caída de la última entrada se enciende.

Para una cabecera esto no es cosmética: un multicast vacío en la salida hace que los receptores y los moduladores se comporten de forma impredecible, mientras que una señal con cortinilla sigue siendo una señal.

El archivo debe estar en esa ruta en **cada** máquina a la que el stream pueda mudarse: la cortinilla la reproduce la máquina en la que el stream se ejecuta en ese momento.

4.5.6 Qué mirar después

Por stream se ve cuántos segundos vivió con la entrada principal y cuántos con las de reserva, cuánto tiempo estuvo sin datos, cuántas conmutaciones y reintentos hubo. Una proporción creciente de tiempo en la reserva significa que la entrada principal es sistemáticamente mala, aunque el aire nunca se haya interrumpido.

Los errores desglosados por causa están en la página [Errores de entrada](#).

4.5.7 Si la reserva se porta mal

- **El aire salta entre entradas**. Suba la ventana limpia: la entrada tendrá que demostrar su salud durante más tiempo.
- **La reserva está en estado «Sin comprobar»**. No está fijado el intervalo de recomprobación — la estación no la visita.
- **La conmutación parece lenta**. Baje el tiempo de espera de fuente. Un umbral demasiado pequeño dispara en falso con cualquier tropiezo de red.
- **No vuelve a la principal**. La principal o aún no se ha recuperado, o no logra completar la ventana limpia — es decir, realmente no está sana.

4.6 Errores de entrada

Un solo contador de errores responde únicamente a la pregunta «hay un problema en la entrada». Para una cabecera eso no basta: la causa decide a quién llamar. Los paquetes perdidos son la red, la conexión rechazada es el socio, el tiempo de espera del handshake es su codificador o el cortafuegos entre ustedes.

Por eso los errores de entrada se cuentan **por causa**, y cada causa es una línea propia en el gráfico y una serie propia en las métricas.

4.6.1 Dos familias de causas

- **Defectos de medios de un stream vivo** — los datos llegan, pero dañados: `lost_packets`, `broken_payload`, `desync`, `ts_pat`.
- **Fallos de la fuente** — no hay datos, y la causa de muerte está nombrada: `connection_closed`, `connection_refused`, `timeout`, `http_request_error`, `not_found`, `denied`, `decode_error`, `protocol_error`, `io_error`, `other`.

La diferencia es práctica. La primera familia significa que la señal corre y hay que repararla a nivel de transporte — lo siguiente que hay que mirar son los [contadores TR 101 290](#). La segunda significa que no hay señal en absoluto, y lo que hay que reparar es la conexión.

Cómo se cuenta:

- la clasificación sale de la categoría tipada del error, y no del texto del log — reformular un mensaje no rompe el contador;
- un fallo al abrir una fuente también entra en el detalle, no solo el fallo de una conexión ya en marcha;
- cada fallo entra también en el contador agregado `errors`, de modo que las sumas cuadran;
- los fallos repetidos en los reintentos se cuentan cada vez: la frecuencia de eventos es exactamente la señal de que «la fuente sigue muerta»;
- las paradas normales — reconfiguración, fin del stream, desalojo por una entrada de mayor prioridad — no se cuentan como errores.

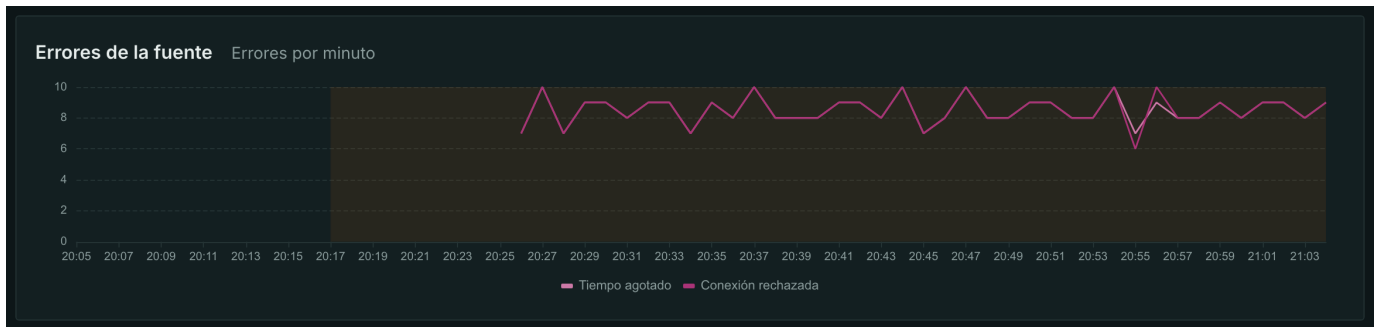
4.6.2 El gráfico en la tarjeta del stream

El panel **Errores de la fuente** en la pestaña **Fuentes** dibuja los errores por minuto como líneas separadas por causa. Las causas sin eventos no ocupan lugar en la leyenda — el gráfico muestra exactamente lo que está pasando.

Los minutos en los que el ritmo de fotogramas de entrada fue cero se resaltan como outage. Eso responde a una pregunta que un solo contador de errores nunca cierra: si aquellos errores eran con la señal viva, o si en ese momento no había señal en absoluto.



Una fuente HLS o HTTP MPEG-TS inaccesible da `http_request_error` en cada reintento, y un handshake SRT que falla por tiempo de espera da `timeout`:



Sin la opción de contadores ampliados el panel muestra una sola línea agregada; las líneas detalladas aparecen junto con la opción.

4.6.3 La serie de métricas

En el servidor Prometheus incorporado el detalle viaja como una serie con la etiqueta de causa — `errors_detail_total{name, cause}`; cuando hay más de una estación, las series llevan también `node`. Las causas sin eventos no crean series.

```
# stream errors over 5 minutes broken down by cause
sum by (cause) (increase(errors_detail_total{name="tv1"}[5m]))

# every input currently seeing connection failures
sum by (name) (rate(errors_detail_total{cause="connection_closed"}[5m])) > 0
```

4.6.4 Cómo leerlo

Qué se ve	Qué significa
<code>lost_packets</code> con bitrate vivo	está perdiendo la red entre la fuente y la estación; use los contadores de transporte para encontrar el PID afectado
<code>timeout</code> a ráfagas, bandas de outage	la fuente no responde; cada ráfaga es un reintento más
<code>http_request_error</code> como peine constante	la fuente HTTP es inaccesible: dirección, DNS, cortafuegos o un origen caído
<code>connection_closed</code> con el stream vivo	el socio corta la conexión él mismo; mire su lado
<code>denied</code> , <code>not_found</code>	acceso o ruta: la clave, el <code>streamid</code> , el nombre del recurso
<code>desync</code> , <code>broken_payload</code>	los datos llegan rotos — el transporte o el codificador del otro lado

Un error suelto en el gráfico es lo normal en una red. El diagnóstico lo pone la **forma**: un peine constante de una causa, ráfagas alrededor de las bandas de outage, crecimiento con el bitrate sin cambios.

4.6.5 Qué viene después

- [Reserva de entradas](#) — cómo convertir el diagnóstico en una conmutación.
- [TR 101 290](#) — contadores de transporte por cada PID.
- [Monitorización](#) — a dónde van estos contadores y cómo alertar sobre ellos.

5. Procesamiento

5.1 Salida MPEG-TS

El receptor del otro extremo no es un reproductor: busca el servicio por su número de programa, las pistas por sus PID y el nombre del servicio en la SDT. Mientras esas tablas no sean las que espera, la señal no existe para él, por correcto que sea el vídeo de dentro.

Todas las salidas MPEG-TS de un stream — la entrega por HTTP, SRT, los destinos por UDP, SRT y RIST — las produce **un solo multiplexor**. La sección **Salida MPEG-TS** en la pestaña **Egreso** define sus tablas de servicio; la misma sección en un destino individual las anula solo para ese destino.

5.1.1 Las tablas del stream

La sección lo dice directamente en el panel: se aplica a todas las salidas MPEG-TS del stream hasta que un push o un SRT play la anule. Los campos vacíos muestran los valores por defecto como sugerencias.

Campo	Qué define	Por defecto
Número de programa (PNR)	la entrada en la PAT, <code>program_number</code> en la PMT y <code>service_id</code> en la SDT	1
PID de la tabla PMT	el PID de la tabla PMT	auto
PID de pistas	PID explícitos de las pistas: una fila «Pista» + «PID» por cada pista	auto
Nombre del servicio (SDT)	el nombre del servicio en la SDT	el nombre del stream
Nombre del proveedor (SDT)	el nombre del proveedor en la SDT	vacío
Transport stream ID	<code>transport_stream_id</code> en la PAT y en la cabecera de la SDT	1
Original network ID	<code>original_network_id</code> en la cabecera de la SDT	1

Las reglas se comprueban al guardar, no en el aire:

- los **PID** están en el rango 32–8190: por debajo está reservado para las tablas de servicio, por encima, para los paquetes nulos;
- los **PID duplicados** dentro de la sección están prohibidos, incluida la coincidencia del PID de una pista con el PID de la tabla PMT;
- el **número de programa** nunca es cero — el número cero en la PAT está reservado para la NIT;
- los **nombres del servicio y del proveedor** juntos están limitados para que la SDT quepa en un paquete TS.

Un PID puede definirse para una pista que ahora mismo no existe: surte efecto cuando la pista aparece. Si un PID explícito choca con el PID asignado automáticamente a otra pista, gana el explícito y la pista desplazada se muda de forma determinista a un PID libre — con un aviso en el log.

MPEG-TS output

Applies to every MPEG-TS output of the stream unless a push or SRT play overrides it

Program number (PNR)
1030

PMT PID
1000

Transport stream ID

Original network ID

Service name (SDT)
TV One

Provider name (SDT)
Example Media

Track PIDs

Track a1	PID 1012	×
Track v1	PID 1011	×
Track	PID	Add PID

5.1.2 SDT

Junto con la PAT y la PMT, la salida lleva una tabla SDT (PID 0x11): un servicio de tipo digital television, `service_id` igual al número de programa, y los nombres de servicio y proveedor tomados de la misma sección.

Las cadenas se codifican según DVB: la latina se envía tal cual y todo lo demás como UTF-8 con la codificación declarada, por eso los receptores leen correctamente los nombres cirílicos.

Al cambiar la configuración, se incrementan las versiones de las tablas PAT/PMT/SDT afectadas y los receptores recogen las nuevas como corresponde. La edición se aplica en caliente: los destinos de entrega y los clientes conectados no se reconectan.

5.1.3 Tablas propias por destino

Un destino individual tiene una sección **Sobrescritura de salida MPEG-TS** con los mismos campos — es la respuesta a una tarea típica de cabecera, en la que la misma señal se entrega a distintos receptores con números diferentes. Está disponible para:

- un **push** por UDP, RTP, SRT o RIST — RTMP no tiene flujo de transporte, por eso allí no hay sección;
- la **distribución por SRT** desde el puerto dedicado del stream.

Los campos vacíos se heredan de la sección del stream — los valores heredados se ven como marcadores de posición. Un valor definido en el destino prevalece sobre el del stream. La excepción es **PID de pistas**: es un único campo, y un mapa no vacío del destino sustituye íntegramente al del stream — no hay fusión por pista.

La salida MPEG-TS por HTTP y el listener SRT compartido del servidor no tienen sobrescritura: siempre sirven la salida compartida del stream.

MPEG-TS output override

Empty fields inherit the stream MPEG-TS section

Program number (PNR): 300

PMT PID: 1000

Transport stream ID

Original network ID

Service name (SDT)

Provider name (SDT)

Track PIDs

Track	PID	
a1	221	×
v1	211	×
Track	PID	Add PID

5.1.4 Por qué esto es barato

Para un destino con tablas propias no se arranca un segundo multiplexor. El destino recibe una copia de la salida compartida en la que solo se reescriben las tablas de servicio y los PID de los paquetes: la disposición de los paquetes en el tiempo, el PCR y los contadores de continuidad del multiplex compartido se conservan.

De aquí se siguen dos consecuencias prácticas:

- las [propiedades CBR de la salida](#) se cumplen también para los destinos sobrescritos — reescribir las tablas no rompe el bitrate constante;
- los destinos con la misma configuración efectiva comparten un stream idéntico byte a byte: cien destinos iguales cuestan lo mismo que uno.

La versión de tablas de una salida reescrita se lleva según su propia historia de cambios, y no se copia de la salida compartida.

5.1.5 Qué viene después

- [Entrega a la red](#) — a dónde va este stream.
- [CBR para DVB](#) — los requisitos del modulador respecto a la tasa y al PCR.
- [TR 101 290](#) — cómo comprobar que las tablas y la continuidad están en orden.

5.2 CBR para DVB

A un modulador no le hace falta una tasa media, le hace falta **uniformidad a nivel de paquetes TS**: el ancho de banda está asignado y el stream debe ocuparlo exactamente, sin caídas ni picos. El mismo requisito lo ponen los receptores profesionales y las redes en las que el transporte se cuenta por ancho de banda y no por una media por minuto.

El CBR en Mcast es una propiedad del multiplexor de salida, no un botón aparte de «activar CBR».

5.2.1 Qué hace el multiplexor

- **La tasa objetivo se deriva de los bitrates de las pistas** — con un margen de alrededor del 5% para el crecimiento y un presupuesto para las tablas de servicio.
- **Los huecos se rellenan con paquetes nulos (PID 0x1FFF)**: el ancho de banda queda ocupado de forma uniforme aunque los fotogramas lleguen a ráfagas.
- **El PCR se coloca desde un reloj sin deriva**, por eso la divergencia entre PCR y DTS no se acumula ni en muchas horas de emisión.
- **Se tiene en cuenta el modelo de búfer del decodificador (HRD)** — los desbordamientos y los vaciamientos se vigilan en lugar de descubrirse en el receptor.

La uniformidad se mide contra el reloj interno del stream (27 MHz), no contra el reloj de pared: la disposición no depende de lo irregularmente que lleguen los fotogramas desde la fuente.

5.2.2 Lo que cuesta la uniformidad

Las tres exigencias de un multiplex — disponer los paquetes de forma uniforme, no detenerse nunca, no perder nada — se satisfacen por pares, pero no las tres a la vez. La uniformidad es una afirmación sobre el futuro, mientras que los datos son un hecho sobre el pasado: para disponer los paquetes uniformemente, la programación tiene que conocer el ritmo por adelantado, y el ritmo solo puede medirse por lo que ya ha llegado.

En la práctica, esto significa:

- **Un PID sin nada que enviar cede su hueco** en vez de detener el multiplex. Una pista de audio anunciada en la PMT pero silenciosa no debe colgar el vídeo — y no lo cuelga.
- **Un PID rezagado se recupera dentro de la ventana** en lugar de ser descartado. Un mux de difusión clásico pierde datos en este punto; aquí el intercambio se resuelve al revés.
- **El ritmo se mide por el PES empaquetado**, no se deduce del ancho de banda declarado multiplicado por una constante.

Lo que de ahí se sigue para el operador: fije el bitrate de salida con honestidad, con margen para los picos, y no al bit justo. Un ancho de banda recortado no es «calidad un poco peor», es un PID rezagado de forma sistemática.

5.2.3 Configure las pistas de salida

Abra el stream, la pestaña **Transcodificador**. Puede empezar desde un preset — por ejemplo **Escalera estándar 1080/720/360 + AAC** — y editar los peldaños a partir de ahí.

En un peldaño del bloque **Vídeo (escalera)**:

- **Bitrate (kbps)** — la tasa objetivo de la pista. La suma de las pistas más las tablas de servicio es el ancho de banda que ocupará el flujo de transporte.
- **Longitud GOP y Estructura GOP** — para difusión, el GOP suele ser de exactamente un segundo: un punto de entrada para el receptor y para el empalme. El botón **Unificar GOP** lo alinea en todos los peldaños.
- **Códec** — con el valor **Tal cual**, el peldaño no se recodifica. Si la fuente ya llegó con la forma adecuada, al multiplexor no le hace falta un transcodificador para disponerla uniformemente.
- **Altura, Rate control y Preset del codificador** — las demás propiedades del peldaño.

El bloque **Codificación** elige quién hace el trabajo: **CPU** o **GPU NVIDIA** (con una tarjeta elegida, o automáticamente la menos cargada).

Warning

Un peldaño passthrough junto a peldaños codificados degrada la escalera: su GOP y sus fotogramas clave no están alineados con los demás peldaños. El panel marca ese peldaño por separado.

5.2.4 Desentrelazado

Una entrada entrelazada — SD MPEG-2, 1080i de difusión — codificada en vídeo progresivo se ve como un peine. El **Desentrelazado** vive en el bloque **Decodificador**, porque es una propiedad de la **entrada** y no de un peldaño: el decodificado ocurre una sola vez y la imagen resuelta va a todas las pistas de salida.

Modo	Qué hace
Automático (solo fotogramas entrelazados)	el valor por defecto: el filtro se aplica a los fotogramas que el decodificador marcó como entrelazados; una entrada progresiva pasa de largo sin coste
Forzado (todos los fotogramas)	para fuentes que mienten sobre la entrelazación
Desactivado	no se construye ningún filtro, los fotogramas van al codificador tal como son

El desentrelazado no cambia la frecuencia de fotogramas: un fotograma progresivo por cada fotograma de entrada.

5.2.5 Comprobación

Qué mirar en la salida:

- **la tasa** en el receptor — no debe flotar; una caída significa que el multiplex no tenía nada que enviar;
- **el jitter del PCR** y los **contadores de continuidad** — véase [TR 101 290](#); un stream uniforme con un PCR incorrecto no le sirve de nada a un modulador;
- **las tablas de servicio** — el receptor debe ver el servicio bajo el número esperado, véase [Salida MPEG-TS](#).

Anular las tablas en un destino individual **no rompe el CBR**: el destino recibe una copia de la salida compartida con PID y tablas reescritos, mientras que la disposición de los paquetes y el PCR siguen siendo los del multiplex compartido.

5.2.6 Qué viene después

- [Salida MPEG-TS](#) — PNR, PID y SDT.
- [Entrega a la red](#) — cómo llega este stream al modulador.
- [TR 101 290](#) — cómo comprobar que la salida sirve.

5.3 Subtítulos y teletexto

Una señal de difusión lleva texto en dos formas: el teletexto europeo (subtítulos en una página, habitualmente 888) y los subtítulos ocultos estadounidenses CEA-608/708 dentro del vídeo. Una cabecera se hace dos preguntas distintas sobre ellos: **sobreviven hasta la salida y hace falta una pista aparte**.

Ambas viven en la pestaña **Procesamiento** de la tarjeta del stream.

5.3.1 Tránsito por defecto

Por defecto, la estación no hace nada con el texto: el PID de teletexto y los subtítulos ocultos dentro del vídeo salen exactamente como llegaron. En el formulario eso se lee así:

- **Teletexto — En tránsito dentro de MPEG-TS (sin pista de subtítulos);**
- **Subtítulos ocultos, modo de entrega — Dentro del video (passthrough).**

Para la entrega a una red ese es el comportamiento correcto: el receptor y el televisor decodifican el texto por su cuenta, y cualquier intervención sería un coste y un riesgo adicionales. El tránsito sobrevive al remux y al reempaquetado en otro transporte.

5.3.2 Cuándo hace falta la extracción

La extracción hace falta cuando detrás de la estación no está un receptor, sino un reproductor: **ningún reproductor OTT decodifica teletexto**. Si de la estación sale una [copia OTT](#), los subtítulos llegarán al espectador solo como pista de texto.

- **Teletexto — Extraer las páginas de subtítulos como pistas:** cada página de subtítulos se convierte en una pista WebVTT. Dentro del MPEG-TS el teletexto sigue viajando: la extracción añade una pista, no quita ninguna.
- **Subtítulos ocultos — tres modos: Dentro del video (passthrough), Pista de subtítulos en su lugar (extract) y Dentro del video y como pista (both).**

El multilingüismo sale gratis: los radiodifusores suelen poner una página de subtítulos por idioma, y cada una se convierte en una pista aparte, sin segunda entrada ni trabajo manual.

El modo extract corta los subtítulos ocultos del MPEG-TS

En el modo **Pista de subtítulos en su lugar (extract)**, los subtítulos ocultos se recortan del vídeo en todas las salidas, MPEG-TS incluido: pushes por UDP, RTP y SRT, segmentos TS y salida por HTTP. Una pista de texto no tiene cómo viajar dentro del MPEG-TS, así que un consumidor del flujo de transporte se queda sin subtítulos en absoluto. Si la señal se toma por MPEG-TS — y en una cabecera ese es el caso habitual —, elija **both**.

La extracción de teletexto cuesta la baja latencia

Una entrada de subtítulos solo se conoce cuando se cierra: mientras la página no cambie, la duración de la línea queda sin decidir. Por eso un stream con pista de texto se sirve como segmentos corrientes, sin segmentos parciales de LL-HLS. Para la entrega a una red eso no cambia nada, pero alarga el borde en vivo de la copia OTT unos pocos segundos — una razón más para no activar la extracción donde la señal solo pasa por la estación.

5.3.3 Qué se ve por stream

Bajo el selector, el panel enumera las páginas de teletexto encontradas en el stream. De cada una se conocen dos cosas independientes, y no hay que confundirlas:

- **declarada** — el radiodifusor declaró la página en las tablas de servicio, con su idioma y su propósito;
- **en emisión** — la página llegó de verdad al stream.

De ahí los estados que verá junto a una página: **declarada pero nunca vista** (el radiodifusor calla), **en emisión pero no declarada** (el radiodifusor olvidó actualizar las tablas — esas páginas se extraen en igualdad de condiciones), o ambas a la vez.

El idioma se muestra junto con su procedencia, porque no merecen la misma confianza: **declarado en el PMT** es un hecho, **heredado de la pista** es una conjetura razonable a partir de una pista de un solo idioma.

La lista se va llenando a partir del aire y solo tiene sentido en un stream en marcha.

5.3.4 El archivo

Los subtítulos ocultos se decodifican y se escriben en el [archivo](#) en todos los modos, así que el modo de entrega puede cambiarse más tarde — los subtítulos aparecerán también en el archivo ya grabado. Las pistas de teletexto extraídas caen en el archivo junto con el vídeo.

5.3.5 Qué viene después

- [La copia OTT](#) — quién necesita la pista de texto.
- [Salida MPEG-TS](#) — qué entra en el flujo de transporte.

6. Entrega

6.1 Entrega a la red

La salida principal de una cabecera es un flujo de transporte enviado a la red: al grupo multicast de un operador de cable, a un modulador, a una dirección unicast de un receptor. Aquí la estación es la parte activa: envía por sí sola, nadie se conecta a ella.

Un destino de entrega se llama **push**, y cada push tiene un **Nombre**. El nombre es su identidad: por él se encuentra el destino en la configuración, en las estadísticas y en las métricas. El panel lo advierte justo debajo del campo: cambiar el nombre crea un push nuevo y los contadores empiezan de cero.

6.1.1 Envíe el stream a un grupo

La pestaña **Egreso**, el bloque **Pushes**, el botón **+ Añadir push**. El protocolo es **MPEG-TS/UDP**.

Campo	Qué define	Por defecto
Nombre	la identidad del destino	—
Host	el grupo multicast o la dirección unicast del receptor	—
Puerto	el puerto del receptor	—
TTL de multicast	cuántos routers sobrevive el datagrama	10
Interfaz de salida	por qué tarjeta de red sale el multicast	la elige el sistema
Dirección de enlace	la dirección local a la que se liga el socket	la elige el sistema

El interruptor **Activado** junto al nombre apaga un destino sin borrarlo. El unicast se diferencia del multicast solo en la dirección: el TTL y la selección de interfaz funcionan igual.

6.1.2 Multicast en una red real

Tres cosas hacen tropezar a la entrega, y las tres se resuelven con los campos de arriba.

- **La tarjeta de red equivocada.** En una máquina con varias interfaces, el sistema elige la interfaz de salida por la tabla de rutas, y para el multicast esa elección muchas veces no es la pensada. Si la señal «se está enviando» pero el receptor no la ve, fije primero la **Interfaz de salida** de forma explícita.
- **TTL.** El valor por defecto de 10 permite al paquete sobrevivir al enrutamiento — el 1 que el kernel trae de fábrica no dejaría salir el stream ni fuera de la subred. A través de varios routers el TTL se sube; para un segmento estrictamente aislado, se baja.
- **El receptor tiene que suscribirse.** La estación envía al grupo sin importar si alguien escucha: los paquetes salen aunque no haya ningún suscriptor. Que «no llegue nada» del lado del receptor suele ser culpa de su propia suscripción o del IGMP snooping del conmutador, y no de la estación.

6.1.3 RTP en lugar de TS a secas

Algunos receptores esperan el MPEG-TS envuelto en RTP según RFC 2250 / SMPTE ST 2022-2, y no un stream pelado sobre UDP. Es la misma salida con otra encapsulación: elija el protocolo **MPEG-TS/RTP** en lugar de **MPEG-TS/UDP**.

El datagrama recibe una cabecera RTP; todo lo demás — direccionamiento, TTL, interfaz, contadores — es común con un push UDP corriente.

La opción vecina en la misma lista es **RIST**, para cuando el lado receptor espera exactamente eso.

6.1.4 Varios destinos a la vez

Un stream tiene tantos pushes como haga falta, y funcionan con independencia: un receptor caído en un destino no toca a los demás.

Los destinos con la misma configuración de salida efectiva comparten **un solo stream idéntico byte a byte**: cien pushes iguales cuestan lo mismo que uno. Lo que los hace distintos son solo sus propias [tablas de servicio](#) — el PNR y los PID propios del destino; entonces para él se reescriben las tablas y los PID, mientras que la disposición de los paquetes, el PCR y los contadores de continuidad siguen siendo los de la salida compartida.

6.1.5 Comportamiento ante el fallo

El pusher no tiene política de reintentos propia: lo revive el stream. Una vez cada cinco segundos el stream coteja los destinos en marcha con la configuración y sustituye un pusher muerto por uno nuevo; el contador de reconexiones cuenta exactamente esas sustituciones.

De ahí la consecuencia práctica: **el envío a un destino inalcanzable nunca se rinde y nunca necesita un reinicio manual**. Seguirá levantándose mientras el destino esté activado.

No cuentan como reconexiones:

- **editar la configuración** — el pusher también se reinicia, pero eso es una decisión del operador, no un fallo del enlace;
- **apagar y encender** — apagar no es borrar: la configuración y los contadores acumulados se quedan, solo se detiene el envío.

Borrar un destino borra también sus estadísticas: los contadores están atados al nombre.

6.1.6 Qué comprobar

Bajo cada push corre una línea de estadística en vivo: el estado — **Enviando**, **Error** o **Desactivado** —, la velocidad en kbit/s, los errores por segundo, las reconexiones y, en uno fallado, también la causa. Todos los campos y las recetas de alertas están en la página [Monitorización de entrega](#).

El estado **Enviando** con velocidad cero significa que no hay nada que enviar: el problema está en la entrada, no en la salida.

6.1.7 Qué viene después

- [Salida MPEG-TS](#) — PNR, PID y SDT, incluidos los propios de cada destino.
- [CBR para DVB](#) — cuándo la señal va a un modulador.
- [Monitorización de entrega](#) — contadores y alertas.

6.2 Entrega por SRT

El SRT lleva la señal adonde el multicast no llega: a otra ciudad, a un socio, a través de internet. Hay dos maneras de entregarla, y se distinguen en **quién establece la conexión**:

- **Push** — la estación llama por su cuenta al destino y hace push del stream. Así se entrega la señal a un sitio que espera una conexión entrante.
- **Distribución (play)** — la estación levanta un listener y espera a que el socio tome el stream. Resulta cómodo cuando la contraparte tiene su propio horario y su propio equipo: usted da la dirección y la clave, del resto se encarga ella.

6.2.1 Envío a un destino

La pestaña **Egreso, + Añadir push**, el protocolo **SRT**:

Campo	Qué define
Nombre	la identidad del destino
Host, Puerto	el listener SRT de la parte receptora
Frase de paso	la clave de cifrado AES; si no coincide, la parte receptora rechaza en el handshake
Stream ID	el identificador por el que el receptor entiende qué le han enviado
Tiempo de espera del handshake (ms), Tiempo de espera del par (ms)	los tiempos de espera de conexión y de inactividad

Una dirección pegada entera como `srt://203.0.113.10:9000` se descompone en los campos, y la **URL SRT** vuelta a montar se muestra al lado — eso es lo que se copia y se entrega a la parte receptora.

Un push muerto se levanta solo, una vez cada cinco segundos, como cualquier otro [destino de entrega](#).

6.2.2 Distribución desde el puerto del stream

El bloque **Reproducción SRT**: active **Permitir pull SRT (reproducción)**, fije el **Puerto** y, a ser posible, una **Frase de paso**.

El socio toma el stream conectándose a `srt://<station-address>:<port>` con su clave. El puerto pertenece al stream por entero — un puerto por stream; un conflicto de puertos entre streams se rechaza al guardar, y no se descubre en el aire.

La distribución desde un puerto dedicado puede tener [su propia sección de salida MPEG-TS](#): el socio recibe la señal bajo el número de programa que necesita, sin tocar los demás destinos.

6.2.3 Qué conviene saber

- **La salida no colorea el stream.** El estado del stream refleja la entrada. Un destino de push caído no pone el stream en rojo — la entrega se comprueba por los [contadores del destino](#).
- **La latencia de la salida es el presupuesto del receptor.** Cuánto tiempo tiene la parte receptora para recuperar pérdidas lo fija ella; la regla de los «tres o cuatro RTT» es la misma en ambos lados — véase [Captura por SRT](#).
- **Las retransmisiones son normales mientras sean pocas.** El contador de retransmisiones de un destino crece cuando el camino pierde paquetes: el transporte está haciendo su trabajo. Lo inquietante no es el hecho, sino la tendencia y su compañía — las reconexiones al lado.

6.2.4 Qué viene después

- [Monitorización de entrega](#) — estado, velocidad, errores y retransmisiones por destino.
- [Salida MPEG-TS](#) — un PNR y unos PID propios para un socio concreto.
- [Entrega a la red](#) — multicast y RTP.

6.3 La copia OTT

Una cabecera no sirve a espectadores — ese es el trabajo de [Catena](#) o de su CDN. Pero a menudo la señal hay que entregarla al mundo OTT directamente desde la cabecera: mostrar al operador la imagen en un navegador, dar una copia a la emisión por internet, hacer push del aire hacia una plataforma.

Esta página trata de esa copia y de sus límites.

6.3.1 HLS y DASH desde la cabecera

Cualquier stream de la cabecera está disponible en:

```
http://<station-address>/streaming/v/<stream>/index.m3u8
http://<station-address>/streaming/v/<stream>/Manifest.mpd
```

Se sirven HLS (incluido LL-HLS con segmentos parciales), DASH y MPEG-TS por HTTP. De fábrica, eso basta para ver la señal en un reproductor y asegurarse de que desde la cabecera sale exactamente lo que debe.

Qué conviene tener en cuenta:

- **Un reproductor necesita sus propios códecs.** Un navegador no leerá un MPEG-2 de difusión — hace falta [transcodificación](#) a H.264/AAC.
- **Un reproductor necesita los subtítulos como pista.** Ningún reproductor OTT decodifica teletexto, consulte [Subtítulos y teletexto](#).
- **La entrega masiva no es tarea de la cabecera.** El balanceo, las zonas, la autorización de espectadores y la contabilidad de sesiones empiezan donde la cabecera termina.

6.3.2 Push por RTMP

Entregar la señal a una plataforma o a un sistema de terceros que espera una conexión RTMP entrante se hace con el mismo push.

La pestaña **Egreso, + Añadir push**, el protocolo **RTMP**. Pide solo un **Nombre** y una única **URL** entera, con la clave de stream incluida: `rtmp://a.rtmp.example.com/live2/<key>`.

- Se admiten H264+AAC, y también HEVC y AV1 en enhanced RTMP; los streams de solo audio también salen.
- Los timestamps en un push empiezan desde cero — tal como los esperan las plataformas.
- Los errores se distinguen en el estado: conexión rechazada, tiempo de conexión agotado, publicación rechazada (la clave está ocupada o es incorrecta), un receptor colgado.
- No hay sección de [MPEG-TS de salida](#) para RTMP: este protocolo no tiene flujo de transporte.

Como cualquier otro [destino de entrega](#), un push RTMP se levanta solo y lleva sus propios [contadores](#): tiene fotogramas, pero no datagramas ni retransmisiones.

6.3.3 Cifrado de la copia OTT

La copia OTT puede entregarse cifrada. El bloque **Cifrado de la entrega (DRM)** en la pestaña **Entrega** del stream activa Common Encryption con el esquema `cbs`: HLS y DASH salen como texto cifrado, y la lista de reproducción y el manifiesto llevan la señalización para **Widevine**, **PlayReady**, **FairPlay** y **ClearKey**. La clave sale de los ajustes o de su servidor de claves por SPEKE.

Cifra la estación — en este reparto ella es el origen. Los streamers que reparten su copia reciben el texto cifrado ya listo y lo entregan tal cual: no tienen la clave y no tienen a quién pedírsela.

Qué cambia esto en la estación:

- **HLS y DASH se cifran**, incluidos LL-HLS y la salida desde el archivo.
- **MPEG-TS por HTTP se niega**. No hay cifrado para él, y entregar ese stream en claro sería sortear la protección: la petición recibe un `403`.
- **Los pushes no se tocan**. Multicast, RTP, SRT y RTMP salen de la estación sin cifrar — se cifra la entrega a los reproductores, no el transporte hacia su propio equipo.

Todo el recorrido puede comprobarse sin un DRM comercial: deje **ClearKey** como único sistema, indique una clave en los ajustes y abra la página del reproductor — la licencia la emite la propia estación. Este modo no protege nada y solo sirve para comprobar.

6.3.4 Qué sigue

- **Monitorización de entrega** — si el push llegó hasta la plataforma.
- **Transcodificación** — llevar la señal a lo que reproduce un reproductor.
- **Catena** — cuando detrás de la cabecera se necesita un servicio de televisión completo con espectadores.

6.4 Monitorización de entrega

Una cabecera no puede preguntarle al receptor qué midió. Todo lo que tiene son sus propios contadores de envío, y se toman **por destino**.

La clave de la estadística es el par «nombre del stream, nombre del destino», el mismo que en la [configuración del destino](#). Esto no es un formalismo: un stream sale a cinco destinos, y la frase «el stream está entregando 40 Mbit/s» no responde a si llega a alguno de ellos en concreto.



La opción «estadísticas de pushes»

La vista de entrega es una opción de licencia aparte. Corta solo los canales de datos: la sección `pushes` y los campos planos `push_*` de la API, las series `stream_push_*` del scrape y del servidor Prometheus integrado. La recolección en sí nunca se detiene, y los contadores siempre van a la telemetría de Retroview — el análisis en Retroview está disponible también sin la opción.

6.4.1 La vista en el panel

En la pestaña **Egreso**, debajo de la configuración de cada destino corre una línea de estadísticas en vivo: estado, velocidad, errores por segundo, reconexiones y, en uno fallido, la causa ahí mismo en la línea.

Nombre
backup-dc

Protocolo
SRT

Activado

Identifica el push: un nombre nuevo crea otro push y los contadores parten de cero

Host
203.0.113.10

Puerto
9000

Stream ID

Frase de paso

Nombre de host o IP. Una URL srt:// pegada entera se reparte entre los campos.

URL SRT
srt://203.0.113.10:9000

Se arma con los campos de arriba: cópiela para un reproductor o un codificador.

Sobrescritura de salida MPEG-TS

Los campos vacíos heredan la sección MPEG-TS del stream

Número de programa (PNR)

PID de la tabla PMT

Transport stream ID

Original network ID

Nombre del servicio (SDT)

Nombre del proveedor (SDT)

PID de pistas

Pista

PID

Añadir PID

Desactivado

0 kbit/s

errores/s: 0.00

reconexiones: 0

Nombre
cdn-main

Protocolo
MPEG-TS/UDP

Activado

Identifica el push: un nombre nuevo crea otro push y los contadores parten de cero

Host
127.0.0.1

Puerto
5500

TTL de multicast

Interfaz de salida

Dirección de enlace

Sobrescritura de salida MPEG-TS

Los campos vacíos heredan la sección MPEG-TS del stream

Número de programa (PNR)

PID de la tabla PMT

Transport stream ID

Original network ID

Nombre del servicio (SDT)

Nombre del proveedor (SDT)

PID de pistas

Pista

PID

Añadir PID

Enviando

4221 kbit/s

errores/s: 0.00

reconexiones: 0

Nombre
youtube

Protocolo
RTMP

Activado

Identifica el push: un nombre nuevo crea otro push y los contadores parten de cero

URL
rtmp://198.51.100.7/live/secret-key

Error

0 kbit/s

errores/s: 0.00

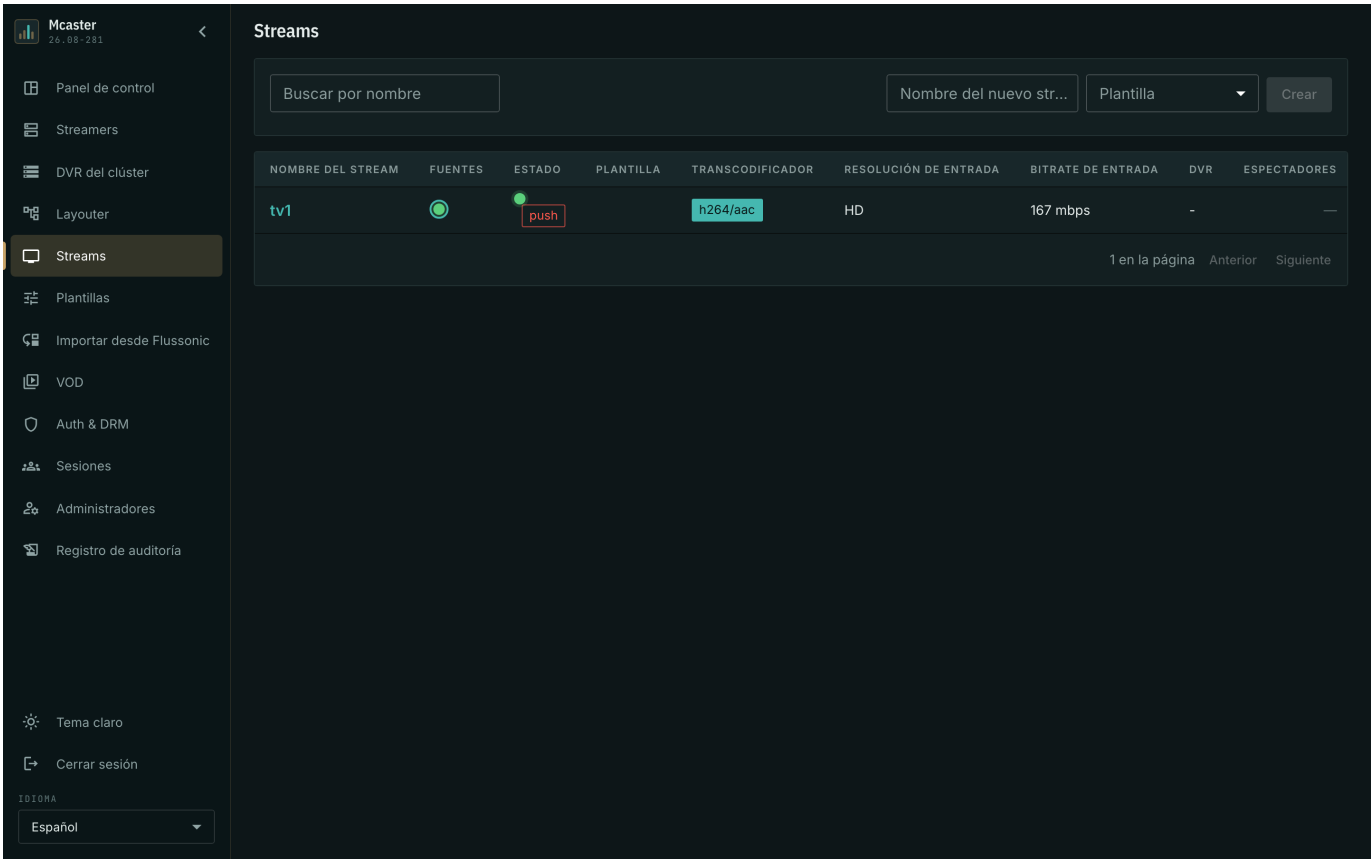
reconexiones: 9

rtmp push youtube init failed: rtmp connect failed: timeout after 5s

En la lista de streams, un destino roto enciende una marca junto al estado del stream: se ve que la entrega está rota sin abrir el stream.

- 52/61 -

© Flussonic 2026



6.4.2 Contadores del destino

GET /streamer/api-v4/streams/stats/{name} trae un objeto indexado por nombre de destino:

```
"pushes": {
  "cable-net": {
    "status": "running",
    "proto": "udp",
    "url": "udp://239.1.1.10:5000",
    "opened_at": "2026-08-13T11:17:53Z",
    "bitrate_kbit": 2113.4,
    "bytes_total": 19053048,
    "datagrams_total": 14478
  },
  "partner-dc": {
    "status": "error",
    "proto": "srt",
    "url": "srt://203.0.113.10:9000",
    "error": "srt handshake rejected: bad passphrase",
    "reconnects_total": 22
  },
}
```

```
"backup-site": {"status": "disabled", "proto": "srt", "url": "srt://198.51.100.4:9000"}
}
```

Campo	Qué significa	Cómo usarlo
<code>status</code>	<code>running</code> , <code>error</code> o <code>disabled</code>	no <code>running</code> durante más de N minutos — alerta; <code>disabled</code> es normal, es un destino que el operador apagó
<code>error</code>	la causa del estado <code>error</code>	lo primero que hay que leer: conexión rechazada, tiempo agotado, handshake rechazado
<code>proto</code> , <code>url</code>	el transporte y el destino tal como los definió el operador	se ve adónde va realmente la señal
<code>opened_at</code>	el momento en que arrancó el pusher actual	se renueva en cada reconexión — el canal cae y vuelve
<code>bitrate_kbit</code>	la velocidad de envío medida	claramente por debajo del bitrate del stream — el envío no da abasto
<code>errors_rate</code>	errores de envío por segundo	crece con la conexión viva — el receptor o el canal se están degradando
<code>bytes_total</code>	bytes enviados, acumulados	conciliar volúmenes por destino
<code>errors_total</code>	errores de envío, acumulados	—
<code>reconnects_total</code>	resurrecciones de un pusher muerto	crece con <code>status: running</code> — el canal se corta y se recupera
<code>retransmitted_packets_total</code>	retransmisiones del transporte (SRT, RIST)	crecen — pérdidas en el camino hacia el receptor
<code>datagrams_total</code> , <code>frames_total</code>	las unidades enviadas: datagramas en los transportes de paquetes, fotogramas en RTMP	—

Los campos en cero se omiten de la respuesta: un destino RTMP no tiene datagramas, uno UDP no tiene ni fotogramas ni retransmisiones.

Junto a los contadores del stream hay tres sumas sobre todos los destinos — `push_bytes_total`, `push_errors_total`, `push_reconnects_total`. Responden «cuánto envió la cabecera en total» sin recorrer el mapa.

6.4.3 Series de Prometheus

Cada contador de destino es una serie propia con la etiqueta `push`:

Serie	Etiquetas
<code>stream_push_bytes_total</code>	<code>stream</code> , <code>push</code> , <code>proto</code>
<code>stream_push_errors_total</code>	<code>stream</code> , <code>push</code> , <code>proto</code>
<code>stream_push_reconnects_total</code>	<code>stream</code> , <code>push</code> , <code>proto</code>
<code>stream_push_retransmitted_packets_total</code>	<code>stream</code> , <code>push</code> , <code>proto</code>
<code>stream_push_datagrams_total</code> , <code>stream_push_frames_total</code>	<code>stream</code> , <code>push</code> , <code>proto</code>

En el servidor Prometheus integrado los mismos contadores están disponibles con las etiquetas `name` (stream) y `push`; cuando hay más de una cabecera, se añade `node`. La profundidad de protocolo — `stream_push_srt_*` y `stream_push_rist_*` — llega junto con la opción de contadores extendidos.

```
# velocidad de envío por destino
rate(stream_push_bytes_total{stream="tv1"}[1m])

# la salida total del stream por todos los destinos
```

```
sum by (name) (rate(stream_push_bytes_total{name="tv1"}[1m]))

# canales que se cortan: reconexiones en cinco minutos
increase(stream_push_reconnects_total[5m]) > 0

# pérdidas en el camino hacia el receptor por SRT/RIST
rate(stream_push_retransmitted_packets_total[1m])
```

6.4.4 Cómo leer el estado

- **status: error con un error no vacío** — no se está enviando nada y la causa está nombrada. El destino sigue levantándose solo, una vez cada cinco segundos.
- **status: running pero bitrate_kbit en cero** — la conexión está, los datos no: compruebe si la señal está llegando por la entrada.
- **status: running con un reconnects_total creciente** — el canal se corta y se recupera; `opened_at` muestra la edad de la conexión actual.
- **status: disabled** — el destino está apagado con el interruptor. Los contadores quedan congelados en sus últimos valores; esto no es una pérdida de datos.

6.4.5 Las alertas con las que empezar

Pregunta	Dónde mirar	Señal del problema
¿Llega la señal?	<code>pushes[].status</code> , <code>bitrate_kbit</code> ; <code>rate(stream_push_bytes_total[1m])</code>	el estado no es <code>running</code> ; la velocidad en cero en un stream vivo
¿Por qué no llega?	<code>pushes[].error</code>	el texto de la causa: conexión rechazada, tiempo agotado, handshake rechazado
¿Se corta el canal hasta el receptor?	<code>reconnects_total</code> , <code>opened_at</code>	las reconexiones crecen; <code>opened_at</code> se renueva cada pocos minutos
¿El camino pierde paquetes?	<code>retransmitted_packets_total</code>	crece con el estado estable
¿La salida da abasto?	el <code>bitrate_kbit</code> del destino contra el bitrate del stream	la velocidad del destino es sistemáticamente menor

6.4.6 Qué sigue

- [Entrega a la red](#) y [entrega por SRT](#) — con qué se configura un destino.
- [Errores de entrada](#) — la otra mitad del cuadro: qué llega a la cabecera.
- [Monitorización](#) — los canales de estadística en conjunto.

7. Archivo

7.1 El archivo

Una cabecera no necesita el archivo por el espectador, sino como prueba: qué fue exactamente lo que enviamos a las 03:40, si había señal en la entrada en el momento de la queja, qué publicidad se emitió en la hora en disputa. La grabación ocurre en la misma máquina que recibe y entrega.

El archivo es un almacenamiento propio: el stream se va añadiendo a archivos por horas, por separado en cada pista, en varios discos a la vez. Leer el archivo no interfiere con escribirlo.

7.1.1 Los discos de la máquina

Los discos se configuran una sola vez en la configuración de la máquina, la sección **DVR**. El panel avisa justo ahí: estos ajustes cambian dónde se escribe el archivo en la máquina.

- **Raíz del archivo** — el punto de montaje, la casa del catálogo y la base de las rutas relativas.
- **Disco 1, Disco 2, ...** — con el botón **Añadir disco**. La ruta de un disco es el nombre de un subdirectorío dentro de la raíz, no una ruta absoluta.
- **Ruta del catálogo** — el directorio de servicio para los metadatos del archivo; un campo vacío significa `<root>/catalog`. Aquí no se guarda el archivo.
- **Comprobar montaje antes de escribir** — conviene activarlo siempre: sin ello, un disco desmontado se convierte en un directorio vacío cualquiera, y la cabecera escribirá el archivo en el sistema de archivos raíz hasta agotarlo.

Sin un solo disco la configuración no se guardará: el DVR necesita al menos uno.

Junto a ellos viven los **discos de caché** — ahí se asienta el archivo ajeno ya leído; la grabación en vivo nunca los elige, y el espacio se libera según lo antigua que sea la lectura de un fragmento. Una cabecera los necesita solo si sirve el archivo de máquinas vecinas.

7.1.2 La grabación de un stream

Abra el stream, la pestaña **DVR**, active **Grabar en DVR** y fije los límites:

- **Retención (horas)** — cuántas horas conservar;
- **Tamaño máximo (bytes)** — el techo de tamaño.

Con ambos fijados, gana el más estricto. La limpieza se ejecuta una vez por hora: la profundidad borra solo las horas plenamente vencidas, el límite de tamaño cuenta desde las grabaciones más nuevas hacia las más viejas.

La grabación se reparte sola entre los discos activos: las horas consecutivas de un mismo stream se alternan entre discos, los streams vecinos se reparten de forma pareja y las pistas de un stream multipista se escriben en paralelo. Un disco pasado a solo lectura se sigue leyendo, pero no recibe datos nuevos — así se retira un disco sin perder el archivo.

7.1.3 La lectura

El archivo se lee por tiempo absoluto: un intervalo, un desplazamiento hacia atrás desde el directo (rebobinado) con unión sin costura con el directo, un preview JPEG del fotograma en un momento dado. Eso alcanza tanto para investigar un incidente como para exportar una grabación a petición.

Si el stream tiene activada la [extracción de subtítulos](#), las pistas extraídas se escriben junto al video.

7.1.4 La comprobación de la grabación

El estado de la grabación se ve en la misma pestaña: la ventana del archivo, las horas con datos y los huecos entre ellas, el tamaño del archivo, las velocidades de escritura de los fragmentos. Qué significan las desviaciones:

Señal	Qué significa
DVR configurado pero no está grabando	el stream está fuera de línea o la grabación nunca arrancó
la frontera del archivo se retrasa respecto a la hora actual	la grabación se detuvo
el límite inferior dejó de avanzar	la limpieza no se ejecuta — más adelante hay un disco lleno
huecos: N h con una ventana ancha	hay agujeros en el archivo: el stream se cayó o la grabación estaba apagada
Errores de escritura del archivo — se pierden datos	el archivo está perdiendo datos ahora mismo, compruebe el disco
Grabación degradada (fragmentos collapsed/skipped/delayed)	los fragmentos se escriben despacio, con retraso o se colapsan — el almacenamiento se está degradando

La última fila es una señal temprana más que una constatación de pérdida: la proporción de escrituras lentas crece antes de que aparezcan los errores.

7.1.5 Qué sigue

- [Monitorización](#) — a dónde van los contadores de grabación.
- [Subtítulos y teletexto](#) — qué cae en el archivo junto con el video.

8. Control de calidad

8.1 Contadores de transporte (TR 101 290)

ETSI TR 101 290 es la lista sectorial de comprobaciones del flujo de transporte: qué debe cumplirse para que un flujo cuente como apto y en qué orden importan esas infracciones. Mcaster lleva esos contadores **por cada PID**, tanto en la entrada como sobre el flujo analizado, de modo que la pregunta «¿degradó la señal o no?» deja de ser cuestión de opinión.

Esto se diferencia de los [errores de entrada](#): allí está la causa por la que una fuente murió o se estropeó; aquí, una medición del transporte mismo.

8.1.1 Qué se cuenta por cada PID

Contador	Qué significa	Qué hacer
<code>errors_ts_cc</code>	una infracción del continuity counter — se perdió o se repitió una porción del transporte	el indicador principal de pérdida; búscala en la red o en la fuente
<code>errors_ts_tei</code>	el flag transport error indicator está activado: el error lo marcó un nodo anterior	el problema llegó antes que nosotros — mire aguas arriba
<code>errors_ts_scrambled</code>	los paquetes siguieron cifrados	no se retiró el cifrado: CAM, llaves, configuración del descrambler
<code>errors_ts_psi_checksum</code>	la suma de control de una tabla de servicio no coincidió	la tabla llegó rota; si esto no deja de crecer, los receptores pierden el servicio
<code>errors_ts_pmt</code> , <code>errors_ts_pat</code>	problemas con las tablas de programas	el servicio no se ensambla; el programa anunciado puede no estar en el tronco
<code>errors_pid_lost</code>	el PID desapareció del flujo	una pista se fue — a la salida eso es silencio o pantalla negra
<code>broken_pes_count</code>	paquetes PES rotos	carga útil dañada
<code>empty_packets</code>	paquetes vacíos en el PID	por lo general es normal, pero junto con una caída del bitrate es señal de inanición

Los errores de todo el flujo — `errors_ts_pat` (PAT ausente o rota) y `errors_ts_service_lost` (un servicio anunciado desapareció) — se cuentan aparte de los PID: significan que lo que se desmoronó no son los datos, sino su descripción.

8.1.2 PCR: exactitud y jitter

El PCR es el reloj del receptor. Si tiembla, el decodificador no sostiene su búfer y la imagen se deshace aunque el flujo esté intacto.

El jitter se cuenta en buckets, con el `PCR_AC` = ±500 ns del estándar como frontera:

- **exactamente cero y hasta 500 ns** — dentro de tolerancia;
- **de 500 ns a 500 μs** — una infracción de exactitud;
- **por encima de 500 μs** — una infracción grave;
- `pcr_resync` — hubo que reanclar el reloj: una rotura de la escala, no un temblor.

No existe adrede un campo único que lo sume todo: un «jitter medio» esconde justamente aquello para lo que existe la medición. Lo que hay que mirar es la distribución — una proporción creciente de los buckets `fail` con un bitrate sin cambios significa que la fuente o el transporte dejó de sostener el ritmo.

8.1.3 El búfer del decodificador (HRD)

`hrd_buffer_min`, `hrd_buffer_max` y `hrd_buffer_underrun` describen el modelo de búfer del decodificador hipotético: si el flujo llega al receptor de manera que el búfer ni se vacíe ni se desborde.

Esto responde a una pregunta que ver el stream en una computadora no puede: **si el flujo sobrevivirá a un decodificador de hardware**. Un reproductor de software perdona lo que una set-top box o un modulador no perdonarán.

8.1.4 La línea de tiempo: saltos y no monotonicidad

Un grupo aparte de contadores va sobre el tiempo: `pts_goes_backwards`, `pts_jump_forward`, `too_large_pts_jump`, `time_corrections`, `corrected_backward_pts`.

Para una cabecera esta es una señal temprana de una fuente mala: un flujo cuya escala salta llegará al receptor, pero el empalme, la grabación y la multiplexación sufrirán por ello constantemente.

8.1.5 Cómo usar esto

- **Una alerta sobre** `rate(errors_ts_cc) > 0` por cada PID caza la degradación del transporte antes de que un espectador la vea.
- **Los paquetes cifrados** son un signo binario: o no hay ninguno, o la configuración está mal.
- **El jitter del PCR y el HRD** son el criterio de aceptación de la salida antes de entregarla a un modulador.
- **El contador CC en la entrada** puede gobernar el aire: el umbral `cc_errors_limit` marca una fuente como degradada y la cabecera se pasa a la reserva — véase [Redundancia de entradas](#).

La profundidad por PID la abre la opción de licencia de contadores extendidos; el agregado de errores está disponible siempre. A dónde van los contadores y con qué mirarlos está en la página de [Monitorización](#).

8.1.6 Los límites

Mcasters mide lo que se ve desde el análisis del transporte. Las comprobaciones que exigen la parte de radiofrecuencia — nivel de señal, MER, BER antes y después de la corrección — pertenecen al hardware de recepción y no a la cabecera, y aquí no se cuentan.

8.2 Monitorización

Mcaster siempre recolecta el conjunto completo de contadores. La licencia y la tarea solo deciden por qué canal los lee usted.

Canal	Qué da	Disponibilidad
El panel y la API	una instantánea: qué hacen ahora mismo el stream y sus destinos	todos
El servidor Prometheus integrado	un datasource de Grafana listo para usar, con unas horas de historia	todos
El scrape de Prometheus	métricas en bruto para su propia TSDB y su alertmanager	la opción de contadores extendidos
Retroview	la monitorización en la nube del fabricante: meses de historia, tendencias, alertas listas	suscripción

La regla de elección es sencilla:

- «¿Qué está pasando **ahora**?» — el panel o la API.
- «¿Qué pasó en las **últimas horas**?» — el servidor Prometheus integrado.
- «¿Qué pasó **la semana pasada** y por qué se cayó de noche?» — Retroview.
- «Quiero **mi propio** Prometheus, Grafana y alertmanager» — el scrape.

8.2.1 Qué mirar exactamente

Los contadores en sí están explicados en las páginas de cada tarea, no aquí:

- [Monitorización de entrega](#) — destinos: estado, velocidad, errores, reconexiones, retransmisiones.
- [Errores de entrada](#) — causas de los fallos y minutos sin señal.
- [TR 101 290](#) — calidad del transporte por cada PID.
- [Redundancia de entradas](#) — tiempo en la principal y en la reserva.

8.2.2 El servidor Prometheus integrado

Dentro de Mcaster corre un servidor Prometheus: la HTTP API `/api/v1/query`, `/api/v1/query_range`, `/api/v1/series`, `/api/v1/labels`, `/api/v1/label/{name}/values` sobre un almacén propio en memoria. Para Grafana es un datasource listo — añada un datasource de tipo Prometheus y apúntelo a la URL de la cabecera. Con esas mismas consultas se alimentan los gráficos del panel.

Límites por construcción:

- la historia es de unas horas, en memoria; tras un reinicio el almacén queda vacío. Es el canal del «qué está pasando ahora», no un archivo de métricas;
- PromQL está soportado como un subconjunto: selectores con matchers de etiquetas, `rate` / `irate` / `increase`, las agregaciones `sum` / `avg` / `min` / `max` / `count` con `by()` / `without()`, aritmética, `offset`. Una construcción no soportada devuelve un error explícito, no un resultado distorsionado;
- el conjunto de series es el básico; la profundidad por PID, SRT y RTP llega con la opción de contadores extendidos.

```
# la velocidad de captura del stream
rate(stream_input_bytes_total{stream="tv1"}[1m])

# errores de entrada en 5 minutos por causa
sum by (cause) (increase(errors_detail_total{name="tv1"}[5m]))

# la salida por cada destino
rate(stream_push_bytes_total{stream="tv1"}[1m])
```

Cuando hay más de una cabecera, cada serie lleva una etiqueta `node`, y un mismo dashboard funciona tanto contra una cabecera única como contra el plano de control: agregue con `sum without (node)` — en una máquina única la etiqueta no está y la agregación no cambia nada.

8.2.3 El scrape para su propia monitorización



La opción de contadores extendidos

Los endpoints del scrape están disponibles solo con esta opción de licencia. Sin ella, use el servidor Prometheus integrado y Retroview.

Métricas en el formato de texto de Prometheus:

- GET `/streamer/api-v4/live-metrics` — streams y entradas;
- GET `/streamer/api-v4/sessions-metrics` — sesiones;
- GET `/streamer/api-v4/dvr/metrics` — los discos del archivo y el catálogo;
- GET `/streamer/api-v4/runtime/metrics` — el proceso y el asignador de memoria.

La historia la guarda usted y solo la limita el retention de su TSDB; está disponible la profundidad completa, incluidas las métricas por PID, SRT y RTP; las alertas las maneja su alertmanager. Los contadores son monótonos y sobreviven a la observación de los reinicios.

8.2.4 Retroview

Retroview es la monitorización en la nube del fabricante. La cabecera envía telemetría una vez por minuto — el catálogo completo de contadores, incluido todo el detalle extendido — y Retroview la guarda durante meses. En la cabecera no hay que configurar nada: el canal funciona junto con la licencia.

Cuándo ir allí y no al servidor Prometheus integrado:

- **historia y tendencias** — el crecimiento de la carga a lo largo de los meses, la planificación de capacidad;
- **post-mortems** — qué hacía el stream de noche: la telemetría sobrevive a los reinicios;
- **conciliación de consumo** — los contadores son muestreados en el tiempo por un sistema externo;
- **análisis de causas sin la opción de contadores extendidos** — la telemetría siempre lleva el detalle;
- **alertas listas** — reglas de producto en lugar de reglas caseras.

8.2.5 Los logs

Los logs son estructurados y el nivel se fija con la variable `RUST_LOG`:

```
journalctl -u mcaster -f
```

Los registros llevan el nombre del stream y el módulo de destino, así que el filtrado por un stream concreto funciona sin un indexador externo. Las trazas se exportan por OpenTelemetry (OTLP).

8.2.6 La salud de la cabecera

- Sonda de readiness: GET `/streamer/api-v3/monitoring/readiness` — para balanceadores y orquestadores.
- Memoria por parte del pipeline: `sum by (part) (stream_memory_bytes)` — un crecimiento monótono sin crecimiento de la carga merece una investigación.