

DOOH: Digital Out Of Home

What is Digital Out of Home?

Digital Out of Home (DOOH) is digital advertising shown on screens in physical, public spaces - think digital billboards along a motorway, screens in shopping malls and supermarkets, displays in lifts and lobbies, transit and airport panels, and menu boards in quick-service restaurants.

It's the digital evolution of traditional "out of home" advertising (printed billboards and posters). Instead of a static printed sheet, content is delivered electronically to a screen, so it can be scheduled, rotated, and updated remotely.

DOOH works quite differently from web or in-app advertising, and those differences shape how the integration below is built:

- **There's no browser, cookie, or logged-in user.** Ads are targeted to a *screen* (and its location), not to an individual person.
- **Content is played, not clicked.** Success is measured by *proof-of-play* - confirmation that a creative actually appeared on screen - instead of clicks or impressions in regular advertising.
- **The network can be unreliable.** A screen in a lift or a remote billboard may lose connectivity, so players are expected to cache content and keep running offline.

Where Adhese fits in

Adhese acts as the **ad server** for your DOOH network. It decides what should play on each screen and provides the building blocks a player needs to run:

- **Media asset distribution:** delivering the video and image files to be played.
- **Playlist generation:** telling each screen or player what to play, and in what order.
- **Proof-of-play tracking:** recording that each creative was shown.
- **Player-level targeting and reporting:** targeting and measuring by individual screen or location.

How an Adhese DOOH integration works

A typical DOOH setup involves the following components working together:

- **Screens:** the physical displays that show the ads.
- **Digital Signage CMS / Player:** the software driving the screens; it fetches content from Adhese and plays it.
- **Adhese Ad Server:** decides what plays and serves assets, playlists, and tracking URLs.
- **Server-side data mapping:** links each player to metadata (such as its store or location) for targeting and reporting.
- **Reporting and Proof-of-Play tracking:** captures what actually played, for measurement and billing.

In short: the **player** asks Adhese for a playlist, Adhese responds with the ads to show, the player plays them from its local cache, and it fires tracking URLs back to confirm each play.

Before you start

Make sure the following are in place before beginning the integration.

Adhese requirements

- An active Adhese account.
- The DOOH feature enabled by Adhese support.
- Configured inventory and templates.
- A configured player ID target.

Player / CMS requirements

Your player or CMS must be able to:

- Perform HTTP `GET` requests.
- Parse JSON.

- Cache assets locally.
- Play back video and images.
- Fire tracking URLs.

The DOOH endpoints

There are two things a player fetches from Adhese:

1. **Assets:** the creative files to download and cache in advance (*optional*).
2. **A playlist:** the list of ads to play (*required*). The playlist comes in two variants: one with a capped number of ads, and one with no cap.

Purpose	Endpoint
Assets download (Heads-up)	<code>https://headsup-[customer].adhese.org/api/headsup/download-list/sl[positioncode]</code>
Playlist - limited number of ads	<code>https://ads-[customer].adhese.com/m/stack/sl[positioncode]?max_ads=[amount]</code>
Playlist - unlimited number of ads	<code>https://ads-[customer].adhese.com/e/stack/sl[positioncode]</code>

Assets Download Endpoint (Heads-up)

The heads-up endpoint is **optional**. Use it to retrieve and cache all active creative assets for a DOOH position ahead of time. The response structure is fixed and cannot be customised.

Request example

```
https://headsup-[customer].adhese.org/api/headsup/download-list/sl[positioncode]
```

Response example

```
{
  "media": [
    {
      "ad": {
        "id": "https://pool-demo.adhese.com/pool/lib/ad1.mp4",
        "mime": "video/mp4",
        "curl": "https://pool-demo.adhese.com/pool/lib/ad1.mp4",
        "checksum": "abc123"
      }
    }
  ]
}
```

```
]
}
```

Detailed information on the heads-up endpoint is available [here](#).

Playlist Endpoint

Adhese provides two **stack** endpoints for retrieving a playlist:

- `/m/stack/`: returns a list with a **limit** on the number of ads (`max_ads`).
- `/e/stack/`: returns a list with **no predefined limit**.

Request example

```
/m/stack/: https://ads-[customer].adhese.com/m/stack/sl[positioncode]/pi[player
ID]?max_ads=[amount]
/e/stack/: https://ads-[customer].adhese.com/e/stack/sl[positioncode]/pi[player ID]
```

Response example

```
{
  "ads":
  [
    {
      "id": "https://pool-demo.adhese.com/pool/lib/562_2nd_1.mp4",
      "dur": 28.07,
      "proofOfPlay": "https://ads-
demo.adhese.com/track/3474/sl357/tlnone/piplayer_id/A2?1746011926824",
      "error": "https://ads-demo.adhese.com/track/3474-
PLAY_ERROR_[ERRORCODE]/sl357/tlnone/piplayer_id/A2/?1746011926824"
    },
    {
      "id": "https://pool-demo.adhese.com/pool/lib/561_2nd_1.mp4",
      "dur": 11.45,
      "proofOfPlay": "https://ads-
demo.adhese.com/track/3444/sl357/tlnone/piplayer_id/A2?1746011926824",
      "error": "https://ads-demo.adhese.com/track/3444-
PLAY_ERROR_[ERRORCODE]/sl357/tlnone/piplayer_id/A2/?1746011926824"
    },
    {
      "id": "https://pool-demo.adhese.com/pool/lib/560_2nd_1.mp4",
      "dur": 18.85,
      "proofOfPlay": "https://ads-
demo.adhese.com/track/3455/sl357/tlnone/piplayer_id/A2?1746011926824",
      "error": "https://ads-demo.adhese.com/track/3455-
PLAY_ERROR_[ERRORCODE]/sl357/tlnone/piplayer_id/A2/?1746011926824"
    }
  ]
}
```

```
}  
]
```

Detailed information on the stack endpoint is available [here](#).

Targeting and reporting

Player ID target

You can add extra targeting information to a request by appending target sections to the URL, in the form `/[prefix][value]/`.

The **player ID** is one such target. It uses the prefix `pi` (as shown in the examples above) and can be used for:

- Player / screen-level targeting and reporting.
- Location-based targeting and reporting.

Providing extra target / reporting data

Player and store location, plus any other custom metadata, can be supplied **server-side** by uploading a **player → metadata mapping** in Adhese.

Each metadata property is linked to its own target prefix. Adhese maps this data automatically to incoming requests based on the player ID, making it available for both targeting and reporting.

The player ID prefix - and any other target prefixes - must be configured by Adhese support.

Operational recommendations

Asset caching

Always cache assets locally to:

- Reduce bandwidth usage.
- Avoid playback interruptions.
- Improve offline resilience.

Offline mode

If connectivity is lost, players should keep playing using:

- The last valid playlist.
- Cached creatives.

Troubleshooting

Issue	Possible cause
Empty media array	No active campaigns, or a targeting-related issue
Endpoint redirects	DOOH endpoints not enabled
Heads-up endpoint empty	No active campaigns, or <i>Use in heads-up</i> format config disabled
Playback errors	Unsupported codec or corrupt file
No reporting	Proof-of-play URL not triggered

Revision #18
Created 2026-07-06 12:47:39 UTC by Kim Van Crombrugge
Updated 2026-07-15 07:47:13 UTC by Kim Van Crombrugge