

Display ads

Display advertising with Adhese can be delivered through several integration routes, ranging from a drop-in web SDK to a fully server-side API. All routes ultimately talk to the same adserver engine, so the choice of integration method is about *how your platform requests and renders ads*, not about what campaigns can run.

Display advertising covers visual ad formats rendered in a reserved container. Adhese delivers all common display creative types through every integration method described in this document:

- **Image banners** - a hosted image with a click-through link; the simplest and most common display format.
- **Rich media** - rich, animated or interactive creatives built as HTML5 bundles and delivered as an HTML fragment.
- **Third-party tags** - ad markup hosted by an external ad server or demand partner. Adhese returns the tag; the creative itself is loaded from the third party at render time.

Where can display ads run?

Anywhere the platform can request ads and render the returned markup or image:

- **Websites** - desktop and mobile web, whether ads are requested in the browser (SDK, tags, Prebid) or server-side by your CMS or backend before the page is delivered.
- **Apps** - retrieve ads using the Request API and render them in a webview or native component.
- **Newsletters** - set up a server side connection to retrieve ads, download the material and put it in the newsletter before sending them out

Most common display formats

- **300×600** Half Page
- **300×250** Medium Rectangle
- **336×280** Large Rectangle
- **728×90** Leaderboard
- **970×250** Billboard — large-format desktop top slot
- **320×100** Large Mobile Banner
- **320×50** Mobile Leaderboard
- **120×600** Skyscraper
- **160×600** Wide Skyscraper

Integration methods

Method	Best for
TypeScript SDK	Modern websites and web apps
Prebid.js / Prebid Server	Publishers running header bidding
Request API	CMS, apps, custom stacks

Revision #4

Created 2026-07-06 15:20:29 UTC by Kim Van Crombrugge

Updated 2026-07-15 07:47:13 UTC by Kim Van Crombrugge