| Section Metadata | |
| --- | --- |
| style | content |



# Upgrading to aem.live from hlx.live

| Deprecation |
| --- |
| End-of-service warning As outlined on our [deprecations and end-of-service page](/docs/deprecation#hlxlive-and-hlxpage), the hlx.live domain will no longer be available after **December 18 2025**. Adobe strongly recommends all customers with production sites still running on hlx.live to complete their migration to aem.live as soon as possible, preferably before **March 31 2025**, to prevent interruptions in site availability. |

If you are familiar with \*.hlx.live for published content and \*.hlx.page for previews, please note that we are introducing \*.aem.live and \*.aem.page as new domains for your sites.

With this change, we are also introducing several improvements while making sure everything works just as before. **The most important benefit is that preview URLs will load faster on aem.page thanks to caching.**

In this guide, we will walk you through the changes you'll need to make to upgrade to aem.live. Many of these steps are only required if you use specific features, so you may not need to complete all steps. Rest assured, the process is straightforward, and we are here to help you every step of the way.

## Try your site on aem.live

Go to your site on hlx.live and replace hlx.live in the URL with aem.live. For instance, if you used to see your site on [https://main--helix-website--adobe.hlx.live/](/) then your new URL would be [https://main--helix-website--adobe.aem.live/](/). No more hlx.live, instead it's aem.live now.

## Try previews on aem.page

If your site works on aem.live, then previews will also work on aem.page. For instance, if you used to see your site on [https://main--helix-website--adobe.hlx.page/](/) then your new URL would be [https://main--helix-website--adobe.aem.page/](/).

## Tell Sidekick to use aem.live and aem.page

Even though both hlx.page and aem.page work for previews now, Sidekick will always default to hlx.page.

Let's change this. In your Sharepoint or Google Drive folder, locate the .helix/config spreadsheet and add following values, replacing <owner with your GitHub organization and <repo> with your repository name:

* cdn.live.host: main--<repo>--<owner>.aem.live
* cdn.preview.host: main--<repo>--<owner>.aem.page

Preview the spreadsheet using Sidekick to activate the new configuration.After activating the configuration, please make sure the Sidekick preview action is opening urls with .aem.page.

## Point your CDN to aem.live

In your [CDN setup](/docs/byo-cdn-setup), change the origin URL or host name from hlx.live to aem.live and activate the change. If you experience any issues, then revert the origin URL or host name back to hlx.live and contact Adobe for assistance.

**We recommend to make the change in lower environment such as stage before performing it in production environment**

**That's it, you made it! Your site is now on aem.live. For 95% of all sites, this is all you have to do. Keep reading for the remaining 5% of special cases.**

**If you need help during the upgrade or ran into issues, please** [**join our Discord channel**](https://discord.gg/aem-live) **or** [**get in touch with us**](https://www.aem.live/business/reachout) **to discuss further.**

# Advanced Upgrade Scenarios

## If you are using BYO DNS without BYO CDN

If you use [custom domains without a custom CDN](/docs/byo-dns), please contact Adobe to perform the origin change to aem.live.

## If you are using hlx.page or hlx.live urls in your code

Check your GitHub repository for any references to hlx.page or hlx.live. If you find them, double-check the functionality, as the changed hostname may introduce CORS or other cross-origin issues. Update the references to aem.page or aem.live.

*Code references to hlx.page and hlx.live in the sampleRUM() function, lib-franklin.js or aem.js do not need to be changed.*

## If you are using hlx.page or hlx.live in your content

Check your content repository for any instances where you might have references for hlx.page or hlx.live in any content areas, including spreadsheets, for project-specific functionality.

Update link references around project documentation areas that may include, but not limited to, *Block Library*, *Runbooks*, *Authoring Guides*, and *Pull Request Templates*.

We don't need to update in-content or absolute links/hrefs pointing to hlx.page or hlx.live in the Word documents, as they will be auto-transformed to aem.page and aem.live.

## If you use the deprecated forms endpoint to receive form posts in Excel or Google Sheets

The [forms service on hlx.live has been deprecated](/developer/forms) and will not be supported on aem.live. Please contact Adobe to discuss possible alternatives for this service.

## If you have configured external services to use hlx.page or hlx.live

Many services like Google reCaptcha, Google Maps, Cloudflare Turnstile, etc. use the site domain or referrer domain to validate API tokens. If you have configured these services to allow your site on hlx.page or hlx.live, make sure to also allow aem.page and aem.live

## If you want to update aem.js, too

Upgrading to aem.live does not require code changes. Some teams use the opportunity to also update to the latest version of [aem.js](https://github.com/adobe/aem-lib). It's best to treat this as a separate upgrade and (like all code changes) test it on a separate branch.

| Metadata | |
| --- | --- |
| name | deprecation |