

Most modern content management systems provide two different ways of editing site content: in-context editing and back-end editing.

While in-context editing is often seen as 'sexier', each method has its strengths and weaknesses. This briefing will explore these two editing options, providing advice on when to use them in practice.

In-context editing

In-context editing allows authors to browse the published website, using site navigation in the normal way to find the desired page.

By clicking a small or hidden button (or some other equivalent action), they can switch into editing mode, updating the content of the page in place.

During editing, the author can see how the published page appears, including the formatting of the text. By updating the content 'in context', the author can immediately see the finished product, even as changes are being made.

Depending on the vendor, this functionality may be called different things, including 'live site editing', 'in place editing' or 'surf-to-edit'.

The big advantage of this method is its simplicity. Authors are familiar with the structure of the published site, and comfortable with navigating through it.

By hiding most of the underlying complexity of the CMS, this allows authors to concentrate on updating the content.

For these reasons, in-context editing is often seen as the more usable authoring option. It is also commonly seen as a more 'modern' option for updating the site.

In-context editing is not without weaknesses. The very simplicity of the interfaces makes some tasks harder, or at least, less obvious.

Updating metadata is often hidden, as there is no natural place for it in an in-context editing interface. Likewise, workflow and other page-level options are less clear.

Even more problematic are tasks such as moving a page or creating a new page. These are very difficult to present without exposing the author to the back-end CMS interface.

The net result is that in-context editing is most effective for small updates of existing pages. It is also a better interface for less experienced or less frequent authors, allowing them to work on the site without being exposed to the full complexity of the CMS.

Skilled or frequent authors can find in-context editing to be inefficient and limiting, and may prefer to work in the back-end environment, as discussed below.

Back-end editing

The traditional way of editing content in a CMS is to log in to the system, and to use the back-end (administrative) interface to update pages.

This typically provides a 'tree view' of the site structure, allowing the author to find the desired page, update content, change metadata, and trigger workflow.

This is the more powerful of the two interfaces, and it gives authors access to the full functionality of the CMS. (Depending on security levels, individual authors may be limited in the capabilities that they are allowed to use.)

This is unquestionably a better interface to use when restructuring the site, creating new site sections, or doing other administrative activities.

It may also be a more efficient interface, better suited for more experienced or regular authors.

It is, however, a more complex interface. As discussed in the article, *11 usability principles for CMS products*, the challenge is to ensure that the back-end interfaces are quick and easy to use despite the functionality offered.

Conclusion

Both in-context and back-end editing have their place, and each targets a different set of users.

Organisations selecting a CMS should research the needs of their authoring community, and then rigorously assess how well the products meet those needs.

In many cases, organisations will be best served by getting a CMS which supports both methods, but usability remains a key challenge regardless of the authoring method used.