

While it is important to identify key functional requirements when selecting a content management system (CMS), some of the most important issues relate to *how* the CMS works, rather than *what* it does.

In particular, the issue of *self-sufficiency* is one that should be addressed by all organisations looking to purchase a content management system.

The issue

Historically, organisations have focused almost exclusively on functional requirements when selecting a CMS. For example, the workflow capabilities of the product, or the publishing features.

In some cases, however, organisations have selected a CMS that matches all the requirements outlined in tender, only to find that the product does not meet their real needs in practice.

In many cases, this is due to a mismatch in expectations of what skills will be needed, or the amount of effort involved.

For example, a product may provide powerful workflow capabilities, but these may need extensive programming abilities to configure. If an organisation doesn't have these skills then the budget can be blown out by expensive consulting, and the project can fail.

Two forms of self-sufficiency

To avoid encountering these issues, organisations should set clear expectations regarding their self-sufficiency in two areas:

- administration
- development

Self-sufficiency (administration)

Organisations are often looking to become more self-sufficient in the administration of their websites. In the context of a CMS, this includes:

- managing users
- creating or changing workflow rules
- restructuring the site
- modifying page templates

If self-sufficiency is required, the CMS should ideally provide a simple user interface in these areas. If programming is required to make these changes, the CMS team must have the skills and knowledge required.

On the other hand, changes to some aspects of the system (such as the page templates) may be very infrequent. In this case, self-sufficiency will be much less of a selection criteria.

Self-sufficiency (development)

Self-sufficiency in development relates to customising or extending the CMS product, beyond the levels offered by the product's tools.

While administrative self-sufficiency is often desired, it is often not needed when it comes to development. In many cases, organisations may simply not have the internal IT resources to make these changes, and would prefer to look to the vendor for support in this area.

There may also be a desire to minimise customisation (to constrain project costs and risks), by looking to purchase an "out of the box" CMS product.

If self-sufficiency is required, then the CMS must be coded in a language that is understood by the internal developers. Open-source products are also a potentially effective approach, as they can provide the greatest degree of self-sufficiency regarding further development.

If self-sufficiency is not required, then the CMS can be treated as "black box". In this situation, it is less important for the CMS to match the IT environment used in the rest of the organisation.

Tender documents

Organisations should decide on their levels of self-sufficiency in these two areas. More importantly, they should be documented, and included in the tender documents.

While this will give the prospective vendors valuable context, further work will be required to assess whether the products meet the identified criteria. In practice, this can often best be done via scenarios (see *Using narrative in a CMS tender*) or by pilot projects.