

14) Content creation

General remarks

On many occasions in this guide I refer to "**content**" as a product component and to "**content creation**" as a project phase or subproject or work package. The objective of this short chapter is to make a few remarks on the subject, in particular concerning the **relationship between content creation and software development**.

The creation of content is often called "**content development**", but I prefer to use the expression "content creation" in order to clearly distinguish it from the development of the software component of a product, which can be described as the "**container**".

Note that when I use the word "development" without qualifying it, I mean "software development", not "content development" (likewise for "develop" and "developer").

The primary purpose of many software applications (including websites) is to make **content** available to end users. Projects of the kind this guide focuses on generally relate to products that feature some amount and form of content.

Content can be "**editorial content**", which globally refers to **text and multimedia assets** (photographs, drawings, maps, animations, videos, sound, etc.), such assets often being used to illustrate text, and sometimes being supplemented with text (eg captions for pictures or videos). It is usually created by people with **editorial skills**.

Content is sometimes simply called "**data**", for example commercial information to be fed into the database of a Customer Relationship Management ("CRM") system, to distinguish it from the "editorial" type of content as described above, although editorial content may also be considered as data.

"**Metadata**", including "metadescriptions", used for content indexing or for the description of illustrations or web pages, should also be considered as content, since it should be written by editorial-type people (as opposed to technical people).

There is also content that may be qualified as "**secondary**" or "**ancillary**", with respect to the "main content" (or "core content") of a product, but which is **nevertheless important**, for example: help text, credits, licencing and other legal information, privacy notice, sales terms & conditions, contact information, etc.

Some products feature no editorial content other than **Help** text (possibly with illustrations) and sometimes **templates** (eg sample presentations), directly available within the product or accessible via an associated website.

Finally, elements of the **user interface** of a product (eg menus, menu items, dialog boxes, text buttons, tooltips, warning and error messages, etc.) may also be considered as **content** insofar as the related text needs to be written by people with editorial skills.

Whatever the amount and form of content to be integrated into a product, the **content** needs to be **created and/or sourced**.

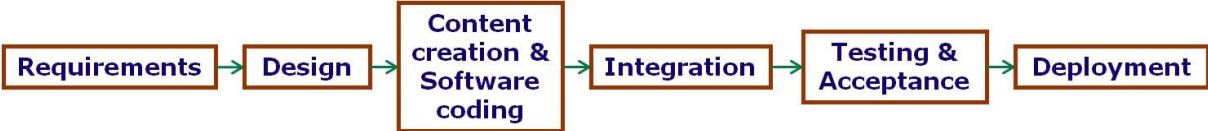
The entity that provides content is often referred to as a "**content provider**", which may be the **project owner**, for example a publisher, and/or a **third party** (possibly an "editorial packager" or an individual contributor) used as a **subcontractor** by the project owner to create and/or source content for a product.

The **content** of a product **is generally not provided by software developers**. Even in the case of a product created by a software development/publishing company (eg Microsoft), content such as Help and templates is not created by software development engineers but by people with editorial skills (which developers generally do not possess...) who belong to an editorial department within the company or who work as or for a subcontractor.

Content providers and software developers need to work in **close cooperation** for the creation of any product featuring content as well as software.

Relationship between content creation and software development

As shown in the following diagram (which also appears in chapter 3, "The life cycle of a project", under "Project execution"), **content creation and coding (implementation) of the software part of a product** can generally be performed **in parallel** (by different people with different skills). These activities are usually featured as **distinct work packages and/or subprojects** in the project plan and organization.



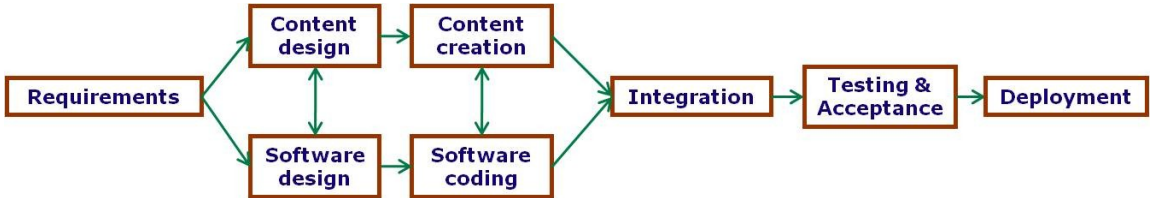
The "Integration" phase in the above diagram refers to the integration of content with the software, the result being a **complete product featuring both content and its container, the software**.

Note that content such as **Help and templates** cannot be created independently of the software since it is **directly dependent on the functions and features of the software**. The creation of such content can nevertheless be done partially in parallel with software development but it cannot be completed before the product is in its final stage of testing and debugging.

As mentioned in chapters 8 and 13, "content" is one of the many subjects to be addressed in a product's requirements and design specifications. Although some of the content design activities may be performed independently of the design of the software, there is a **strong dependence between the two areas of design**.

For example: software developers need to know enough about what a product's content will be in order to establish how it will be stored (possibly in a database), retrieved and displayed; developers also need a detailed description of how the content will be structured and tagged, as well as of categories of metadata that will be attached to content items (refer to the section "Data delivery format specifications" in chapter 13, "Design specifications").

The following diagram illustrates more precisely than the one above the relationships between **content design & creation** and **software design & coding (implementation)**.



Some content providers do not have any **"data engineering"** resources, so they lack the knowledge and skills to define and implement an adequate **data structure and format** for their content. In such a situation, developers may be required to do the necessary work in addition to actual software development. They may even be required to develop a set of **tools** to more or less automatically structure (or restructure) and format content as documented in the design specifications.

Some elements of content, for example **animations**, may require **software development work**. In such a case, editorial people will design and write the scenario of the animations, and the programming will be done by specialized developers, who may be different from those developing the product's main software. Creating content such as animations is usually treated as a "**sub-subproject**" in a project's plan and organization.

Content delivery

At some stage in the development process, **sample content** will need to be integrated with the software, so that developers can check that content complies with the data delivery format that has been specified and that content-related software functions (store, index, search, retrieve, display...) work properly.

Sample content is often required by developers **very early in the development process**, sometimes even at the design stage, and certainly before the first "**Alpha**" version of a product is built, at which stage a **content subset** may be provided.

The **complete and final content** of a product should in theory be provided to the developer before the first "**Beta**" version of a product is built.

One **exception** to this rule is **Help** content, which should perfectly match all functions and features and user interface of a product as they appear in its final version. Help is therefore usually "frozen" in its final state just before the **final version** of the product is completed (which is often a challenge in terms of timing and project management...).

Content updates

The content of some products, in particular websites, needs to be enhanced and updated on a more or less frequent basis.

As mentioned in the "Product evolution" section of chapter 4 ("The life cycle of a product"), the initial project undertaken to create such products should include the design and development of a "**back office**" (or "back end"), which is a set of tools built to enable people in charge of creating, enhancing and updating a product's content (editors, etc.) to do so without requiring any intervention from technical specialists such as software developers.

Obviously, the back office needs to be **designed in close cooperation with its future users** so that the tools that are developed perfectly meet the editors' needs and make the task of content addition and updating simple and efficient.