# Elements of an XML-Based Publishing Workflow

It is useful to think about the publishing workflow as a linear process with several stages that happen in order. Of course, reality is often more complex, and in fact one of the key benefits of setting up an publishing workflow is that these stages can often happen *simultaneously*—one stage does not have to be finished in order for the next stage to be started—and *dynamically*—work can flow from stage to stage as the project demands. Nevertheless, for the purpose of explanation is it useful to think of an archive publishing workflow as a roughly linear process with the following stages:

[1. Writing and Editing](#_Toc463898561)

[2. Curating Content](#_Toc463898562)

[3. Designing Products](#_Toc463898563)

[4. Composing Ebooks](#_Toc463898564)

[5. Typesetting Pages](#_Toc463898565)

[6. Distributing Publications](#_Toc463898566)

[7. Direct Bookselling](#_Toc463898567)

[8. Engaging Readers](#_Toc463898568)

## Writing and Editing

The first and most obvious part of a publishing workflow that most people will think of is the writing tool: What software will people use to write and edit content? This should be a simple question, but unfortunately it is not. The answer depends on a number of factors:

What platform(s) will the writers and editors be using? Will they be working on files in the local filesystem or on content in the cloud? Will they be using Windows, a Mac, an iPad, an Android devices, or (most likely) a combination of the above? The answer to this question will depend both on author and editor preference and on the rest of the requirements of the project.

* Will authors and/or editors need to work collaboratively/interactively? The traditional approach to content creation is to email files around,[[1]](#endnote-1) but this quickly becomes unwieldy and hard to manage as more people become involved in a project.[[2]](#endnote-2) And if content creation needs to happen simultaneously or in real-time, or if the project is more collaborative with no single “hub” editor, then the email method is completely unusable; the project will need to have a web-based content system.
* What capabilities are needed in the authoring/editing environment? Will the authors and editors be creating footnotes and endnotes? How about embedded index entries and hyperlinks to other sections/documents? Or will they only be writing running text in paragraphs? How complex are the formatting requirements for the content that is being created? Do they need to make and edit tables? What about revisions — are they happy just having edits and revisions applied, or do they want redlining that they can review and accept or reject?

Currently, there are some conflicts among these requirements in the marketplace. For instance, if you need real-time collaboration and a moderate amount of WYSIWYG formatting, you and your team will probably work in Google Drive documents, because this is currently the best available real-time collaboration platform. But Google Drive lacks a lot of the document structuring features of a traditional desktop word processor. Or your project might require working collaboratively, though not in real-time, but you also require the use of embedded index entry codes, hyperlinks, content metadata, and complex formatting. In that case, you will probably need a system that enables you to collaboratively share word processor documents through a cloud-based system.

## Curating Content

Content curation is the next step after writing and editing — the manuscript is finished, now it has to be stored in a form that can “feed” the rest of the process.

Content curation can happen dynamically and automatically from early in the writing and editing stage. This enables us to consider, for instance, making it available to readers while it is still being created — just as Danish educational publisher [Systime](http://systime.dk), publisher of over 60% of Denmark’s high school textbooks, has been doing with the textbooks it has recently been developing.[[3]](#endnote-3) You might not need to make your manuscripts available to readers early in the writing and editing process, but there are other advantages of putting all content into curation from the very beginning.

### Advantages of Early-Stage Content Curation

* Designers can begin to work on product design as soon as some of the content is available.
* Content creation and review can happen in a collaborative fashion without bottlenecks.
* Multiple revisions of content can automatically be stored for later retrieval and comparison.
* “Rough cut” or “Beta” ebooks can be produced easily and efficiently at any point in the content development process.
* Reader engagement can become a part of the development process, helping shape the final product.
* Content curation enables an iterative, interactive, dynamic development process without artificial barriers. A traditional, linear process is still possible, of course, but it is no longer necessary as the only possible approach.

When we speak of content curation, we are generally envisioning the storage of content in an archival, future-proof form in a centralized content archive or repository that serves as the permanent “home” of that content and the “hub” of the publishing workflow. We will be discussing several different ways that publishers can create such a content curation hub, and the technologies that enable the other elements of the publishing workflow to be “wired into” the hub.

As with the writing and editing stage, there are questions a publisher needs to answer in implementing content curation:

What technologies will be used to provide the core services? A publishing workflow is built around creating a shared publication content archive, usually stored on a server that provides revision tracking and content validation. There are several ways to provide this kind of system, including licensing (expensive) commercial products. However, we currently recommend that publishers build their content archive on one of two freely available version control systems: *Subversion,* the most widely-used centralized version control system, or *Git,* the most widely-used distributed version control system. Everything that we will want to do with a publishing workflow can be done with either of these two systems. Both are mature and very widely used, and are well within the capabilities of either your in-house information systems team or readily available service providers.

* What content structures will we use to cover the range of what we publish? Most books follow a fairly standard set of structures (parts, chapters, front and back matter), but some kinds of publications have unique requirements. For example, Bibles are very complex and not well-represented by the part-chapter structure of books. In order to enable the publishing workflow to function dynamically, content structures need to be mapped out explicitly and embodied in a set of documents called *schemas,* which are descriptions of the structure of each type of content and publication.
* What policies will be implemented to encourage or enforce the correct content structures for products? The content curation system is able to ensure, by a process called *validation,* that all content and publications have the correct structures. You will want to determine to what extent and in what situations you want to enforce those requirements. For example, you might decide that certain areas of the content archive are exempt, or that certain kinds of work-in-progress do not need to meet the strict requirements of your content schemas.

## Designing Products

Graphic design is primarily concerned with images and layouts. Designers create the artwork (both interior artwork and the cover), the stylesheets, and (usually) the typesetting templates used to produce the product.

Product design doesn’t have to happen at the end of, or separately from, the content development process, as in traditional linear or “siloed” publishing workflows. Instead, product design can happen in a very fluid and dynamic interaction with the content development process when the the content is being managed/curated all along the way. When the boundaries between content development and product design are minimized, new creative potentialities can be unleashed.

## Composing Ebooks

One of the strong advantages of an XML-based publishing workflow is that ebook composition can often be automated, often from very early in the product development process.

Once there is some content, the workflow can include ebook composition, and the output engine can automate the ebook build process.

This, of course, requires having a robust ebook build engine. That’s not a given! The requirements for ebooks are in constant development, and different reading platforms support different features and standards.

An ebook essentially consists of:

one or more content files, formatted for the ebook specification (usually based on some flavor of HTML);

* a “spine” that indicates the reading order;
* publication metadata (publisher, title, author, publication date, etc.); and
* a cover image

Although the details differ among the various ebook format specifications, these are the essential elements. We will be discussing in greater detail what an ebook build engine needs to include and how to put it together.

## Typesetting Pages

Advocates of XML publishing workflows will often talk about being able to automate the typesetting of products. In theory, this is a great idea, as it has the potential to create great efficiencies. In practice, the circumstances under which typesetting can be fully automated are quite limited: If you are creating a lot of publications that follow the same design and content specifications (a large series of books that have the same template, or database-driven publishing), then full automation might be achievable in a particular case. In such a situation, the curated XML content will obviously feed the typesetting process, and final pages can be the result.

A more common and likely scenario is that automation is partial. But even this can provide great efficiencies: Well-structured content flows into the layouts and reduces the hand-work required by the typesetter. We have experienced many cases in which we were able to provide great labor savings through the use of consistently structured content and established templates. We will be talking about some of these case studies along the way.

A difficulty arises as soon as hand-work is introduced into the typesetting process, which is that changes to the content that are made during the typesetting process will need to be brought back into the curated content so that the reality of “one canonical source” can be maintained. XML workflow advocates will sometimes insist that content only flow “one way,” from the archive outward. But it doesn’t have to be that way — it is possible to create a workflow in which content can flow back from typesetting (“roundtripping”). There are a number of situations in which this approach is desirable. We will be talking about how to determine when this is beneficial, and how to set it up.

All of this assumes, of course, that the publisher is using an interactive page layout program such as InDesign — in fact, InDesign is the only such tool that we will consider, because it almost completely dominates the desktop layout world. But there are non-interactive typesetting systems; we will discuss one, TeX, which is widely used in scientific publishing. We will find that even a non-interactive typesetting system does not necessarily yield fully automatic typesetting — tweaks will probably still need to be made by hand.

Under both systems, we will discuss several of the trickier issues in page layout, such as generating indexes, bibliographies, and tables of contents.

## Distributing Publications

Ideally, the publishing industry would have a smooth, automated system by which publishers could let the world know what they have published, and then printers and ebook distributors and retailers could automatically download up-to-date products (the titles themselves) and information about them.

Such a system for distributing products themselves does not yet exist, but there is a widely-used system for distributing product *metadata*, the data about products. This system, called ONIX, enables distributors and retailers to keep up-to-date with what publishers have available.

With regard to distributing products digitally, some of the retailers and distributors who sell ebooks and POD titles do have systems that can be automated, but there is no universal standard for these systems, so it usually requires creating a procedure for each of them. And unfortunately, some of the most important retailers and distributors (Amazon, Apple) have the most cumbersome systems, requiring manual intervention to add, upload, and update titles.

We will discuss how you can get your own system set up, so that you can distribute your products as efficiently as possible. We’ll discuss the current state-of-the-art, with regard to distributing both product metadata and the products themselves.

## Direct Bookselling

Traditionally, a publisher had to send their books out to stores where people in each community could buy the books locally. Now, people in most communities can get to an online store anywhere on the internet. There is no reason, therefore, that a publisher cannot sell their books directly to customers from their own online store, and there are several advantages to doing so. We will discuss these advantages for the both the publisher and the reader, and the technical issues in setting up an online store and connecting it to the publishing workflow (so as to make books available for sale early and efficiently).

## Engaging Readers

One of the benefits of an automated system, such as XML publishing workflow, is that content can be published “early and often” without an undue burden on the publisher. Here, we will focus what that means for the readers, and how to foster their engagement. We will discuss various ways in which publications can be made available to readers early, how their interest and involvement in the composition process can be engaged, and ways to build a community around each book.[[4]](#endnote-4)

1. It is worth reflecting on a situation in which the common practice of five and ten years ago should be considered “traditional.” [↑](#endnote-ref-1)
2. As the managing editor of a major study Bible project who used this method to move manuscripts between over 50 authors, editors, and reviewers, let me tell you, it’s unwieldy and time-consuming. Perhaps that experience more than any other has stimulated my interest in finding and/or creating better solutions. [↑](#endnote-ref-2)
3. Søren Peter Sørensen, “Service Changes Everything.” Tools of Change 2013 presentation. Systime (<http://systime.dk>), Aarhus, Denmark. [↑](#endnote-ref-3)
4. These topics might not seem to be part of a discussion on XML publishing workflow, but in fact, when a book is published “early and often” and directly, reader engagement is part of the publishing process that the publisher manages, and the same sorts of technologies that are used to develop and distribute the book can be used to engage readers. [↑](#endnote-ref-4)