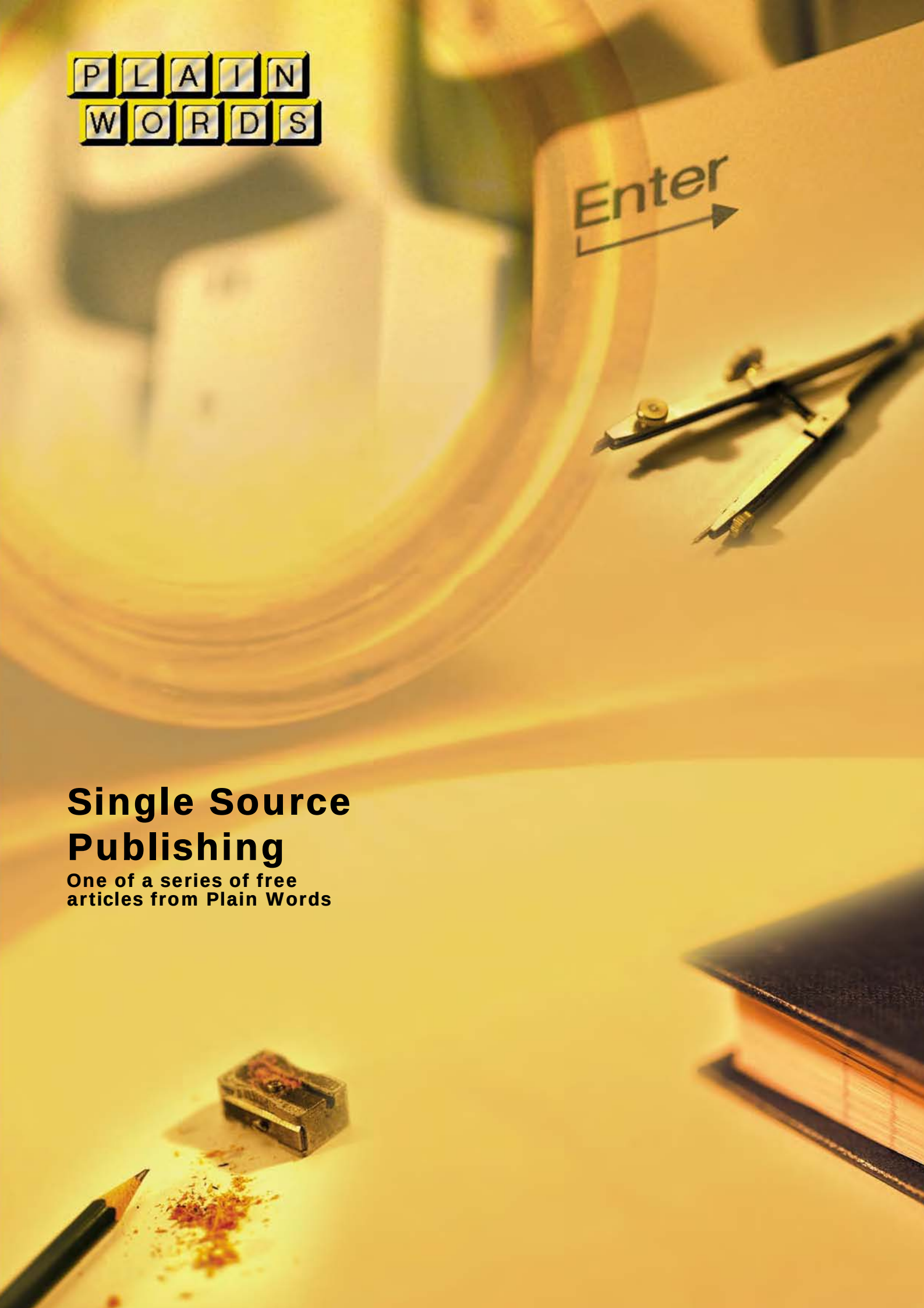

Single Source Publishing

One of a series of free articles from Plain Words

Single Source Publishing

... an Overview of the Pros & Cons

By Peter Meherne

Plain Words' Director

Technical communicators commonly create documentation in a variety of publishing formats and for different purposes, but where the primary content is much the same. An example is creating paper-based user guides accompanied by online Help. The content can be put together in one format and then converted to the other. A specific problem with this approach is that you end up with multiple document source files, which all need to be maintained in parallel with one another. Also a straight conversion of the material may mean that the content of the new format is not ideal for its intended purpose.

Single source publishing provides a solution to this problem. Rather than converting the same information into different formats, it involves collecting all the required information up front and organising it - then developing projects from a single source. A common method used in single source publishing is XML (eXtensible Markup Language), which is derived from SGML, and is related to HTML (for more information see XML tools at the bottom of this article). A more simplistic approach with a limited use is provided by the "single source" options included in a number of authoring tools, such as RoboHelp.

Traditionally, documentation is produced in files that consist of chapters and various sections. These files are then assembled to create a document, such as a user guide or reference manual. Single source materials, on the other hand, are created from one structured document e.g. the XML file. All the information types required by a project (whether it's a user guide, training manual, or series of case studies) are generated from this single file.

Let's take a closer look at the process.

In structured documentation, writers break information down to basic level. If they were compiling information about a person, the various aspects of their life would be split up into categories. So if the person's name were "Joe Bloggs", his name would be assigned under different headings and meanings, related to specific parts of his life and career.

For example:

Book Author - "Joe Bloggs' Guide to Java", published 2000. Best selling reference book on Java applications, etc, etc

Family Tree - son of Alf and Ethel Bloggs

Z-Byte.com - Marketing Director of leading Fortune 1000 software company, etc, etc

Dot.com Bomb - former Webmaster of a failed dot-com start-up

Email Address - joe.bloggs@nowhere.com

Once this is done, writers compile all the elements into a single source file. Within this single source file, they can identify the elements required for each information type. For example, "Book Author" and "Z-Byte.com" might be identified for use in press releases, product brochures, and biographical details.

You would do the same if you were compiling technical information. Each aspect of a product would be categorised, then identified as relating to either training materials, user guide or online Help (or it might be identified as relating to all three).

The important thing to remember is this...

Regardless of the purpose or target audience, all information is stored in one source, the XML file, and can then easily be deployed into your chosen format or structure without affecting the original source.

Integration

One of the major advantages of single source publishing is it brings the whole process of documenting a product together. Literally everyone involved can input their materials - and that includes the marketing and PR divisions. So besides the manual and Help information, you can include marketing materials, newsletters, press releases, speeches, and product brochures, in the single source file.

Another big advantage of single source publishing is it reduces errors when information is updated. Instead of having to update several separate documents, writers only update the single source file from which the documents are generated.

On the downside, though, single source publishing does involve cost - particularly during the setting up stages. Costs can include:

- Re-organising, restructuring, and redesigning existing materials
- Retraining staff in the principles and techniques of single source publishing
- Acquiring and learning how to use new tools

Changing Roles

Another issue surrounding single source publishing is it involves a change in the roles of people within a documentation team - particularly the writers. This is because the process of creating single source materials separates the creation of the input (content) from the output (media or information type).

This can be very positive, however, because writers can spend more time actually writing, rather than getting too involved in dealing with Help tools and related pieces of software.

Single source publishing also expands writers' horizons. Because single source information is used in multiple media (e.g. paper, Help, Web), writers are now responsible for writing information for all those media simultaneously. Therefore, skilled writers need to get an even greater understanding of how the elements of information they create will work in each of the media.

Conclusion

Single sourcing your information has a lot going for it. It can save you time and money, and arguably will improve the quality of your materials. Plus it frees writers from the "mechanical" work of information development and enables them to focus on what they do best - writing.

But it will be a long time before single source publishing becomes the norm for everyone. Many professionals need to learn and develop new skills. All of which takes time and money. Not only that, but existing documentation would have to be reorganised - a massive job.

Despite these drawbacks, single source publishing will eventually become commonplace - after all, for long years it was the "philosopher's stone" of everyone involved in documentation production.

If you would like to relate your experiences of single-sourcing or have any tips please email me at: feedback@plainwords.co.uk

XML Tools

The computer language used in single source publishing is called XML (eXtensible Markup Language). It's a descendent of SGML and is related to HTML.

Like in HTML, you use "tags", or angle brackets (e.g. "<>") to write XML. But rather than giving formatting instructions, such as <b> for bold, or <i> for italic, you give a description of each chunk of data included in your XML file.

If you were writing a CV in XML, you might structure it like this:

```
<cv>
    <candidate>James Lipton</candidate>
    <present_position>Data Manager</present_position>
```

And so on.

This example gives the basic gist of how XML works compared to HTML. But XML is a very complex language and a good deal more exacting than HTML. There's no room for syntax errors, for instance. A file that is not well formed will create what is dubbed a "fatal error", which means that applications will refuse to process the file.

Developers, however, can check for errors using a "parser". Parsers are applications that examine XML code and highlight errors. One good one is called Lark and is available free from...
http://www.textuality.com/Lark/

Indeed, various tools are currently available for XML developers - all free to download from the Web. At a basic level, you can use a Plain Text editor like Window's Notepad to create XML files. As stated above, the process is similar in many ways to writing HTML documents, only you head each page with the tag <?xml version="1.0"?> rather than <html>. And you save each file as ".xml".

If you would rather use a tool built for the job of developing XML, you can download XML Notepad from Microsoft. Go to...
http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnxml/html/xmlpaddownload.asp

Microsoft describe XML Notepad as "a simple prototyping application for...rapid building and editing small sets of XML-based data." XML Notepad gives a tree-view of your document with icons representing text, attributes and comments.

For a list of other XML editors and tools, go to...
http://www.xml.com/pub/pt/3

For an extensive overview of XML, go to the above site's home page...
http://www.xml.com

Or visit the World Wide Web Consortium (W3C)... http://www.w3.org/

And if you're willing to pay £35.99, then buy the following book...

"The XML Handbook"
By Charles F. Goldfarb, Paul Prescod

Goldfarb was one of the inventors of SGML, the markup-language from which XML is derived. So he's an authority to be trusted. And, apart from anything else, the XML Handbook is incredibly extensive and well worth the money if you're looking to develop XML.

Plain Words – The Big Picture

Technical Writing & Course Development

Planning, writing and producing Help systems, HTML, on-line material, traditional manuals. Designing and writing bespoke courses for your products and services, whether tutor-led, CBT or e-learning. Presenting 'train the trainer' or end user courses.

Specialist Search & Selection

Helping you to recruit permanent or contract documentation staff:

- Technical Authors & Editors
- Trainers
- Localisation Specialists
- Information Managers
- Documentation Managers
- Copy Writers
- Project Managers
- Project Coordinators
- Instructional Designers
- Bid & Proposal Writers
- Web Content Developers
- Knowledge Managers

Writing Skills Training Courses

- Structuring & Writing Reports
- Designing & Writing Help Systems
- Writing Winning Bids & Proposals
- Writing Effective Letters & Emails
- Designing & Writing Technical Documents
- Indexing On-line & Printed Material
- Estimating & Planning Technical Documents
- Writing for the Web

Bids Consultancy & Training

"Excellent service, highly skilled authors. All deadlines have been met!"
Giovanni Calamida, European Patent Office

Bidworker, a division of Plain Words, is dedicated to providing bids consultancy and training. Writing, editing and formatting of bids and proposals. 'How to Write Winning Bids and Proposals' training course. Plain Words own **bid**worker™ software to automate much of the process of compiling proposals.

Catalogues

Plain Words **catalogue**worker™ enables you to create high quality printed or online catalogues in minutes. This is a simple and speedy way to create large catalogues that would otherwise need to be built, fully typeset and formatted. It also caters for multi-currency, multi-languages and different discount schemes.

Publicity & Copywriting

Inject selling power into your communications with Plain Words' advertising and PR services. Writing or editing sales copy, web content, articles and newsletters. Publicity campaigns and promotion via the media or through your website.

Other Services

Graphic Design • Printing • Localisation & Translation

Technical Writing • Training • Search & Selection • Bids Consultancy
Call us on +44 (0)1635 202013