

Online Education @ UCC Symposium

Educating the Editor: eJournals and the eJourney

Peter Flynn

Electronic Publishing Unit
UCC Computer Centre

11/11/11 11:11:11

Abstract

The technology for creating and maintaining electronic journals has been around for well over two decades, but it has typically required extensive foreknowledge of programming, or has involved a substantial commitment to a publisher's proprietary commercial software. Editors (both staff and student) spend more time learning how to run the system than actually editing articles.

This paper summarises the pilot phase of a system designed to separate the IT aspects from the editorial and pedagogical, and provide a way for wordprocessed articles to appear on the web and in print with an absolute minimum of technical intervention by the author or editor. The lessons learned from this phase are being implemented in a production system for e-publishing in UCC.

1 Historical background

An electronic journal (eJournal) is generally regarded as a scholarly journal created, processed, and made available in digital form; and usually accessed via a computer network. There is a much wider definition of simply being 'available over the Internet' given by (Cole, 2004) (attributed to the Colorado Alliance of Research Libraries), but Chan (1999) considers narrower definitions, including Lancaster (1995), '[a journal] *created* for the electronic medium and available only in this medium' (my emphasis). We adopt this in the sense of computing technology having been used throughout every stage in its production —

it is incorrect and misleading to use the term ‘eJournal’ to apply to scanned or digitally-reprocessed journals which were originally created by hand or with typewriters, using metal or photographic type-setting, and printed on paper.

Before the days of pervasive networks, there were in fact several corporate technical journals produced on computer which were circulated to clients on magnetic tape (often along with software updates).¹ Traditional publishers of academic journals typically had no access to networking at the time (even if they had had the technical capability), although their readers in selected academic institutions had access to the Arpanet (forerunner of the Internet) from the early 1970s. General access for scholars in other institutions remained for the time extremely limited.

In the early 1980s, with the expansion of networking via Usenet (News), CSNET, and EARN/BITNET, access became available to many more universities and users.² There was a growth of informal serial publications produced by and for communities of scholars with access to the networks, and even students were able to produce online magazines (one of the longest-running, Brent C Britton’s scurrilous *NutWorks*, survived five years from 1985 to 1990). However, most publishers remained offline, although there was a thriving business later in the decade in journal databases distributed on floppy disk and CD-ROM (Bailey, 1990).

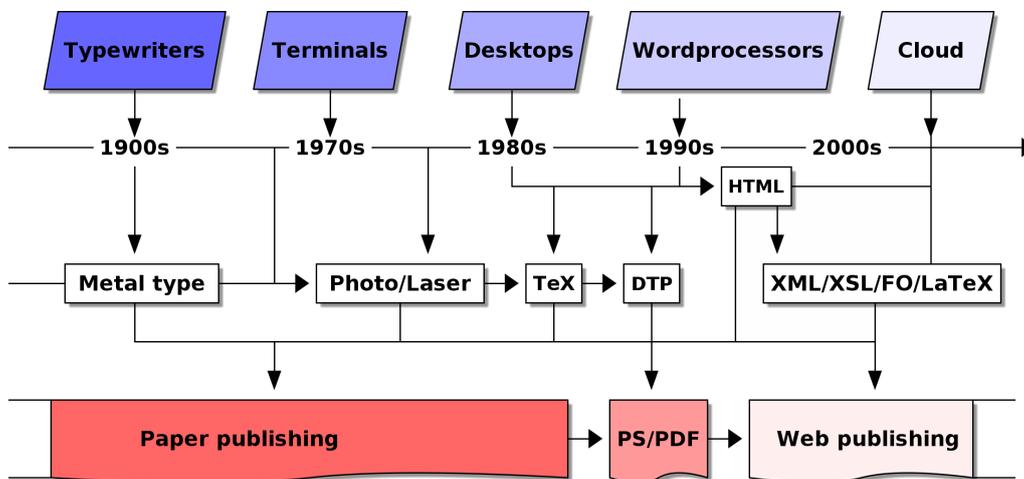


Figure 1: Timeline of technology and journal publishing

At this period, articles themselves were almost always in plain-text (ASCII), as this was easily transmissible over the networks. The plain-text format sometimes also included markup (formatting instructions)

¹Both IBM and DEC produced client journals this way, and the author recalls statistical papers being distributed on tape by P-Stat, a supplier of statistical software, as early as 1983.

²UCC joined EARN in 1984, was a co-founder of HEAnet at the same period, and was connected to the Internet in 1990

such as *roff* or *Runoff* (forerunners of wordprocessing), or the more sophisticated \LaTeX typesetting code, as the only widespread reusable common formats.³ Postscript-formatted output (forerunner of PDF) was possible but the file sizes were usually too large for slow networks and small hard disks. SGML (forerunner of XML) was standardised in 1985, but was used mainly for print-generation until its most popular application (HTML) arrived with the invention of the World Wide Web at the very end of the decade. Conventional peer-reviewed [electronic] journals began to be considered towards the end of the decade— Harter (1998) mentions ‘feasibility studies’, including the Electronic Information Exchange System (EIES) teleconferencing system.

By the 1990s the position had changed significantly, with commercial Internet access, faster university access, and the growth of the Web providing connectivity to a wider population. Scholars in the Natural Sciences and Mathematics had traditionally had greater computer and network access than their colleagues in the Humanities: Harter (1998) claims the first to include graphics was the *Online Journal of Current Clinical Trials* (OJCCT, now available only as scans) in 1992, but there were also some notable ‘firsts’ in the Humanities (Chan, 1996), such as *Postmodern Culture*⁴ and the *Bryn Mawr Classical Review*⁵ (both 1990).⁶ Articles could now be served in HTML and PDF as well as plain-text, although the formatting quality of browsers fell far short of that provided by \LaTeX . An interesting retrospective of a journal still using \TeX (*Solstice*, 1993) appears in Arlinghaus and Zander (2008), whose authors also highlight the problems of citation and the simultaneous publication of both print and network versions.

Since about 2000, almost all publishers solicit, accept, and publish in both print and electronic format, although the ongoing debate over copyrights, accessibility, intellectual property, profit, and open access which restricts online availability seems unlikely to be solved in the near future.

The problem of file formats also persists, as authors are generally unaccustomed to anything except Microsoft *Word* (and unwilling to change). The unsuitability of such wordprocessor formats to publication means a persistent large and expensive conversion task for publishers to produce typesetting as well as Web files. It is little known among authors that the first task of the publisher, once a paper or book has been accepted in its final form, is to rip out all the author’s careful but essen-

³For a long while, plain-text was the only format which could be sent via Usenet News or mailing lists (LISTSERV appeared in 1986).

⁴http://www.press.jhu.edu/journals/postmodern_culture/

⁵<http://bmcr.brynmawr.edu/>

⁶This period also saw the introduction of the Text Encoding Initiative SGML format as a way to preserve and exchange literary and historical texts.

tially useless formatting, ‘right down to the bare metal’, and replace it with the publisher’s own house style.

The levels of access to the Web and to more sophisticated software now mean that self-publishing has also grown, both in scholarly and ‘vanity’ publications. Despite this, a significant level of technical knowledge is still required to install and run a journal-publishing server. ‘Hosted’ services (where the Web software is provided ready to use) are an attractive solution, but tend to suffer over time from ‘bloat’, as eager developers vie to add new and more powerful facilities at the expense of simplicity and ease of use. There is also a risk that a hosting company or its software might go out of business and cease to be available.

2 Rolling your own

The problems attendant on the use of wordprocessor formats for publishing are well-known and evident to anyone who has had to use a wordprocessor for serious writing. However, in 2005–2006, the Organization for the Advancement of Structured Information Systems (OASIS) codified the Open Document Format (ODF) standard, based on XML and implemented by the OpenOffice suite of office software, and this has now become an official international standard, ISO/IEC 26300:2006 Open Document Format for Office Applications (OpenDocument). Not to be outmanœvred again, Microsoft hurriedly paid for extra attendees at ISO meetings to push the vote through for their own competing ‘standard’, OOXML, which is the format of *Word*’s .docx files.

The effect was that, at a stroke, both main wordprocessors now produced reprocessable XML instead of the barely-reusable binary formats that most wordprocessors had used for the previous 40 years. Software to convert XML (of any type) to HTML for Web publishing, and to PDF for print, already existed in several forms, making it technically possible to produce an electronic publishing system that would allow authors to stay using their wordprocessors, and publish articles with a minimum of editorial intervention. However, this rather Utopian view omits a key condition already familiar to anyone who has had to edit another’s work: how to ensure consistency.

2.1 Pilot objectives

The primary objective of developing a eJournal system was to make it easier for editors to bring an issue to publication without the complexi-

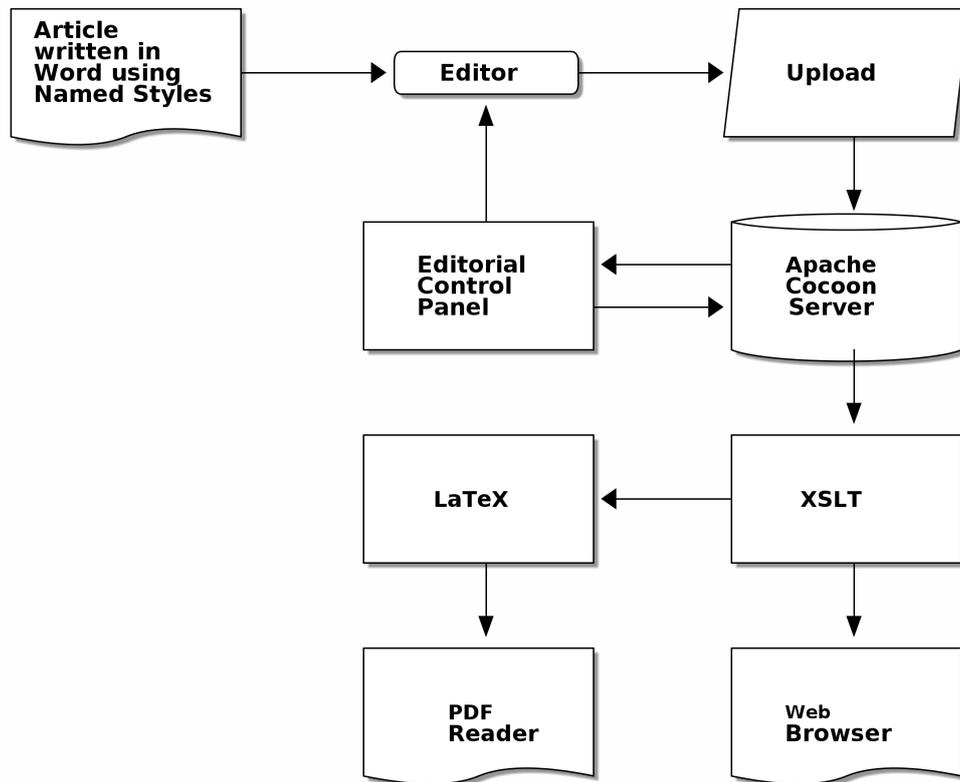


Figure 2: Schematic diagram of XML publishing process

ties inherent in existing systems. The first two eJournals were *Scenario* and the *Irish Journal of Public Policy*.

'Editors' was presumed to mean 'UCC academic and research staff', but with the proposal for *The Boolean* as a postgraduate journal reflecting annual PhD projects, we realised that there was scope for using the system as one of the tools of learning for postgraduates, both for authors and editors. While an eJournal system cannot replace the fundamental editorial tasks, we felt it should remove those mechanical tasks that a computer excels at. Human editors still have to find authors and articles, check the content, proofread, place the figures and tables, manage corrections, and above all, ensure consistency.

From lessons learned in the earlier phase, the editorial control panel (where editors upload articles and test or publish them) was completely rewritten and the process of preparing the document for publication substantially streamlined (Figure 2).

The eJournal system was piloted using the research Web server running Apache *httpd*, the most popular server software, and Apache *Cocoon*, an add-on server for converting XML to web pages (HTML) and other formats on the fly. A Web script (interface program) lets the ed-

itor upload *Word* or *OpenOffice*⁷ documents, test them, and publish them transformed into HTML web pages (articles), using the XSLT styling and processing language (a companion to XML). A second *post-hoc* transformation is done using the same language to create publication-quality PDFs using L^AT_EX.⁸ The pilot was originally scheduled to run for a year, but it rapidly evolved into both a service and a teaching tool which has now been running — not without hiccups — for five years.

2.2 Results: dangerous waters

The astute reader will already have noticed that the control of consistency is a key requirement.

Wordprocessors allow an author to style and restyle anything to look like anything, and to put anything anywhere. By default, there are no controls whatsoever on any formatting or positioning: the wordprocessor interface is completely permissive and purely graphical, as it was designed solely for humans to interpret (whether on screen or in print).

Wordprocessors therefore cannot offer any guarantee that an author will actually put a title on an article, nor give a name, nor provide an Abstract, section headings, figure or table captions, or indeed any descriptive indications of any part of the document. Even if they do, it will only be visual, formatted to the author's own tastes, and while a human can probably interpret them for what they are meant to be, a computer cannot do so unaided. A computer needs accurate and unambiguous labels to say 'this is the title', 'this is the author', 'this is a second-level heading', and so on.

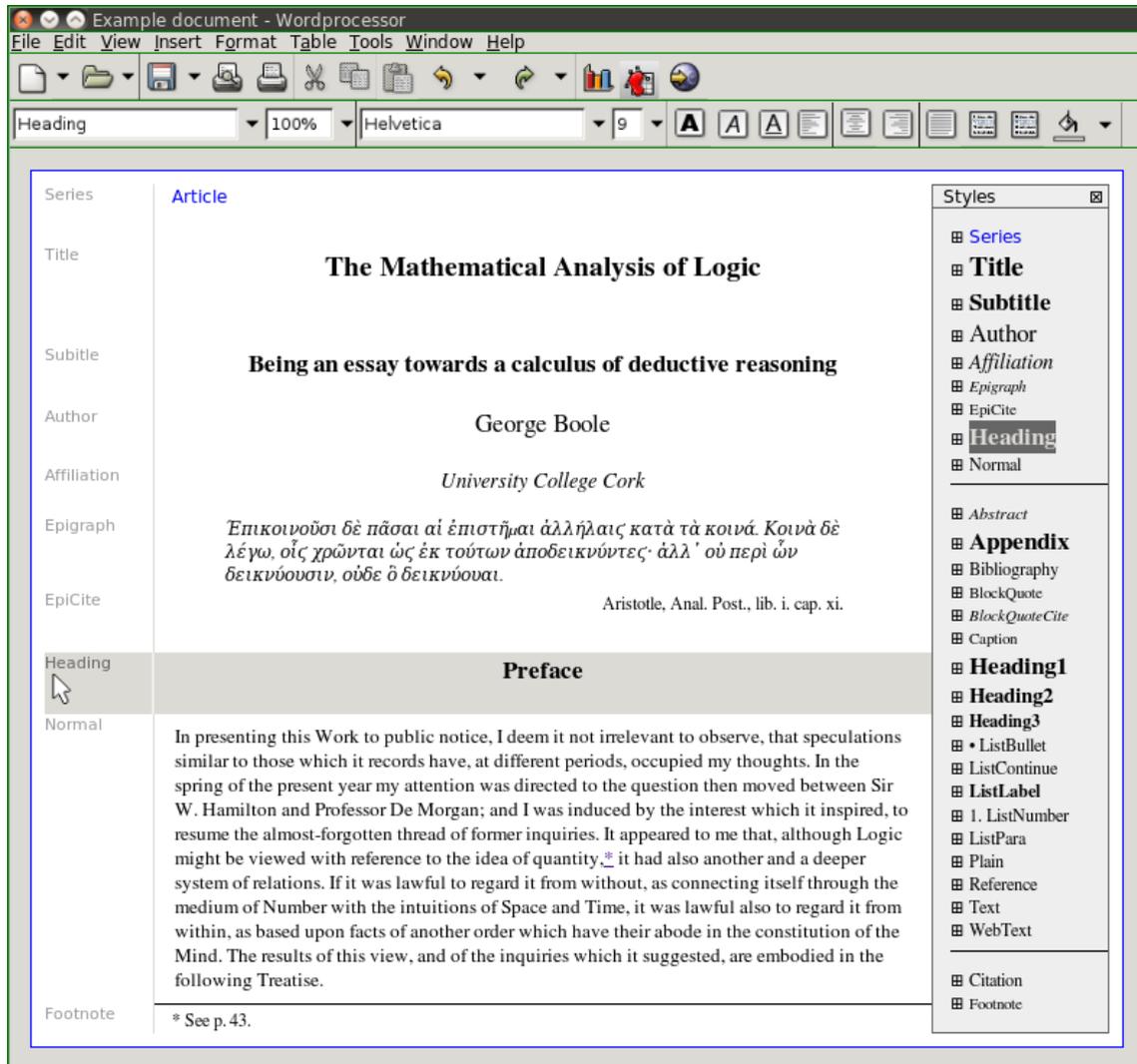
Fortunately, wordprocessors *do* provide a way to do this, called Named Styles. Using these, authors do not need to do any manual formatting; instead, they can just click in a paragraph, click on the relevant style name from a menu, and the paragraph will be styled automatically. More importantly, it will be labelled internally with a meaningful name, like Title, Author, Heading2, and so on. Even better, journal editors can make up their own styles and put them in a template, so that authors can use them to write their articles.

Styles are easy to implement and easy to use (see Figure 3 on the next page): the problem is getting authors to use them. For reasons unknown, styles are not taught in classes on wordprocessing, and writers

⁷Since the acquisition by Oracle of Sun (*OpenOffice*'s progenitor), there is concern over the independent future of the *OpenOffice* suite, so we have moved all development to *Libre Office*, a clone (fork) which is now the default implementation of the ODF format.

⁸L^AT_EX was chosen over the competitor XSL:FO for aesthetic and technical reasons.

Figure 3: The style margin (left) and style menu (right) while writing or editing an article



seem to be unimpressed by the time saved by using them. The marketing activities of the large wordprocessor vendors have been spectacularly successful in the last 20 years or so in convincing writers that so long as the document *looks pretty*, it must therefore *be right*.

In fact, as a little reflection will show, a document that can *only* be interpreted by a human cannot reasonably be processed by a computer. To manage consistency, we therefore insist on absolute adherence to a set of Named Styles. This does not mean restricting the æsthetic activities of a journal or of an author — the editors can specify any number of styles, provided that can be represented sensibly on screen or PDF — merely that, once the styles are specified, the authors must stick to them, and not go down the blind alley of inventing their own, or of using unnamed styling like arbitrary font changes for no apparent reason.

Both *Word* and *Libre Office* come with a rudimentary set of built-in Named Styles, but journals need to add some more in a template, so that the authors can represent their ideas meaningfully. A list of the defaults is in Table 1, but journal editors are free to add and remove from this list (authors are not).

Table 1: Common styles used in eJournals

Name	Description
Series	eg 'Review', 'Editorial', 'Opinion', etc
<i>Paragraph styles</i>	
* Title	title of the article
* Subtitle	immediately following the Title
Author	(one or more)
Affiliation	(one or more, in pairs with each Author)
* Date	date of publication
Abstract	paragraphs of abstract
Epigraph	brief quotation at the start
EpiCite	for an attribution to the Epigraph
* BlockQuote	indented quotation
BlockQuoteCite	for an attribution to the BlockQuote
* Header	unnumbered section heading
* Heading1	numbered section heading
* Heading2	numbered subsection heading
* Heading3	numbered subsubsection heading
Appendix	appendix heading
* ListBullet	bulleted item
* ListNumber	numbered item
ListLabel	labelled item
* ListContinue	second or more paragraphs to an item
* Normal	normal paragraph
Figure	containing one or more images
* Table	standard tables
* Caption	before a Table but after a Figure
Bibliography	entry in the References
<i>Inline styles</i>	
Citation	inline citation
* Footnote	standard footnotes (not citations)
* Emphasis	usually italics

(* = built-in styles)

The resulting *Word* or *Libre Office* documents are uploaded by the editor and set to 'test' status in a control-panel Web page accessible only to the editor[s]. This makes them immediately available via the Web but not in any issue, so they are not visible to an external reader. Any corrections can then be made to the original files, uploaded again, and

re-viewed. When all articles are present and correct, setting them all to a 'published' status immediately makes the issue available to the world.

The only restriction apart from adherence to the styles is that the article documents must be named according to a fixed rule,

nn-aaaa-yyyy-ii-ll.docx (or .odt)

where

nn determines the ordering in the issue (00 is an editorial, 01 is the first article, 02 the second, etc, with 99 used for back matter);

aaaa is the primary author's surname (with an initial if there is a clash, and a digit if still a clash);

yyyy is the volume number (usually the year);

ii is the issue number; and

ll is the language (en is English, de is German, ga is Irish, etc).

The appearance of the journal articles is determined by the journal's editorial board: the default is based on the UCC Web site standard, but a journal is free to employ a designer for something different, provided it can be implemented in HTML or PDF (via the programs described below).

As the textual content of the article documents is all in XML, it can be transformed with 100% reliability into a Web page using a program written in XSLT, the companion styling language to XML. The program uses a list of the styles for that journal, along with logic that determines how they get transformed into HTML. A small file of information about the journal is used to specify the titling, description, frequency of appearance, and other details which vary from journal to journal. Static information about the journal (such as its goals, the names of the editorial board members, a standing call for papers, copies of the style files, etc) are maintained separately using UCC's standard content management system.

PDF-formatted output is generated separately, but it uses another version of the same XSLT program to create typesetting instructions instead of Web page ones. These are passed through the \LaTeX typesetting program, which creates a single PDF file for each issue. This is then copied and split up into separate PDF files for each article, thereby maintaining the page-numbering of the whole-issue file, as the page numbers are still required by many readers for their own referencing systems.

A downloadable reference is provided for all articles, so that readers can capture the citation directly into their personal database (eg *Zotero*, *Mendeley*, *Jabref*, *Endnote*, etc). Issue tables of contents pages provide a complete list of citations for the issue in the same formats.

3 Conclusions

The pilot phase of this project was completed in 2009, with the establishment of the second journal done this way. The project has successfully demonstrated the viability of the process, and fulfilled the criteria of simplicity of operation and the avoidance of specialist technical knowledge by authors, but there is a host of improvements which need to be made. Its use as a tool for teaching writing and editorial skills has yet to be evaluated, and feedback on this is welcomed.

No eJournal system can remove the burden of editing from the editors — student or otherwise. Editors must still deal with gathering ‘final’ versions from reluctant authors. Editors must check articles many things: suitability of the content; the logic of the arguments; the accuracy of references; the pace, depth, and tone of the writing; the spelling, punctuation, and grammar; and a dozen other criteria that editors are expected to deal with. The only addition to this burden is that each article must conform to the ‘house style’ of the journal, by using only the styles prescribed by the editors.

The pilot also revealed a number of technical problems, which are in the process of being addressed:

1. Many authors use obsolete or near-obsolete computer systems and wordprocessors. These can produce an assortment of strange and wonderful approximations to any non-Latin characters that an author may require, such as a pilcrow (¶), the *punctum delens* or *séimhiú*, or the mathematical use of a Greek η . Instead of using the now-universal UTF-8 character set, these machines and programs use anything from relics of the 1970s (the DEC Multinational Character Set) to the vaguely persistent but obsolete Microsoft Windows-1252 Character Set. Editorial conversion to OOXML or ODF is not always successful in these cases, meaning the characters have to be removed and replaced before re-uploading.
2. A similar case obtains with authors who are unfamiliar with computers struggling to represent a character they know exists, but which they are unable to locate in the repertoire available to them. One German author used a Greek β in place of the German ß, presumably for this reason.
3. When an author interrupts a numbered list by interpolating a normal (full-width) paragraph or more of discursive material, the the numbering must be maintained across the divide. The method by which OOXML maintains tenancy between the point of suspension and the point of resumption is astonishing (and in this author’s view, unreasonably) complex, and it is still not certain that the current program handles this correctly.

4. The current system handles standard format pictures (JPG or PNG) by reproducing them with the width and height supplied by the author (or as modified by the editor). Non-standard images such as diagrams drawn in *Word's* internal drawing system (Windows MetaFile) cannot be used on the Web, as they are restricted to Microsoft users only. Until 2010, they had to be converted by the editors, by enlarging the image to as big a size as would fit on a screen, taking a screenshot, saving it as a JPG or PNG file, and then replacing the original drawing by the screenshot. This led to a much poorer-quality image, but was unavoidable at the time. The latest version of *Libre Office* can now convert such WMF files to PDF via a printer driver under server program control, and thence to PNG images: this saves editors much time, and results in better images.
5. A common misunderstanding by authors is that pressing the Enter key to force a line-break (eg in a long title) does *not* insert a line-break: it creates a new paragraph. There are then *two* Title paragraphs instead of one.
6. There is no easy way to honour the attachment of text to images as labels ('call-outs' in publishing-speak) because the geometry of the page changes on the Web (and in the PDF, where figures and tables may need to float to a different location).
7. The user interface (the screen that the reader sees) needs to be simple, but still contain all the information the reader of an article needs. Some useful work by Vilar and Žumer (2005) will inform changes to the interface in the coming year.
8. The proposed assignment of Digital Object Identifiers (DoIs) to articles brings with it a contractual requirement that the References in each article also carry a DoI, where known, and that this is represented as a hypertext link. The second criterion is easy to do; the first less so, and yet another task for editors.

In the current financial and staffing situation there is no easy way to plan for expansion or the acquisition of more resources, but the recent moves to emphasise UCC's commitment both to the Teaching & Learning effort and the dissemination of research activity indicates that eJournals have a role to play.

Acknowledgements

The author is grateful to the many people, editors and readers alike, who offered help and suggestions during the development of the project, and to the NDLR for its financial support.

References

- Arlinghaus, A. L., & Zander, R. H. (2008, Dec). Electronic journals: Then and now... A fifteen year retrospective. *Solstice: An Electronic Journal of Geography and Mathematics*, 19(2). Available from <http://www-personal.umich.edu/~copyright/image/solstice/win08/>
- Bailey, J., C.W. (1990, Dec). Two Electronic Serials on BITNET: The Public-Access Computer Systems Review and the Public-Access Computer Systems News. *Digital Scholarship*.
- Chan, L. (1999, March). Electronic journals and academic libraries. *Library Hi Tech*, 17(1), 10-16. Available from <http://www.emeraldinsight.com/journals.htm?articleid=861177&show=html>
- Chan, L. K. (1996, Sep). Exciting Potential of Scholarly Electronic Journals. *CAUT Bulletin*, 43(7), 9. Available from http://www.cautbulletin.ca/en_article.asp?SectionID=1074&SectionName=News&VolID=262&VolumeName=No%207&VolumeStartDate=9/1/1996&EditionID=29&EditionName=Vol%2043&EditionStartDate=1/1/1996&ArticleID=2647
- Cole, L. (2004, Dec). Back to Basics. *The Serials Librarian*, 47, 77-87. Available from <http://eprints.whiterose.ac.uk/29/>
- Harter, S. P. (1998). Scholarly communication and electronic journals: An impact study. *Journal of the American Society for Information Science*, 49(6), 507-516. Available from [http://dx.doi.org/10.1002/\(SICI\)1097-4571\(19980501\)49:6<507::AID-ASI3>3.0.CO;2-X](http://dx.doi.org/10.1002/(SICI)1097-4571(19980501)49:6<507::AID-ASI3>3.0.CO;2-X)
- Lancaster, F. (1995). The evolution of electronic publishing. *Library Trends*, 43(4), 518-27.
- Vilar, P., & Žumer, M. (2005, Apr). Comparison and evaluation of the user interfaces of e-journals. *Journal of Documentation*, 61(2), 203-227. Available from <http://www.emeraldinsight.com/journals.htm?articleid=1465022&show=abstract>