“If there are documents you really care about: Print them out!” (Vint Cerv, 2015)

Bernd KULAWIK
[a] *Stiftung Bibliothek Werner Oechslin, Einsiedeln, Switzerland*

Abstract. With theses words (only “documents” substituted for the original “photos”) Vint Cerf, one of the ‘fathers of the internet’ and now Google Vice President, warned in 2015 that all our photos – and obviously, documents and research data, too – might disappear soon and that our century may become the “Digital Dark Age”. To avoid this, Cerf is working on a solution named “digital vellum”: It shall provide a platform that can preserve any documents, the software used to create and work with them, the operating system needed for this software and even an emulation of the appropriate hardware. But it may take quite some time before this platform will be available. In the meantime, the good old paper is the only medium that surely can and will survive more than 50 years – the maximum now expected for simple formats like .txt and .pdf files. Even Microfilms (also not usable without technical means) may not survive more than 200 years. But how do we print out digital documents created for and by research: short miscellanea, articles and papers, collections of them and monographs, and in recent years even facebook postings or twitter messages? We write these documents in a dedicated (text) program, sent them to the publisher, who may forward them after several transmissions forth and back with the author(s) to a layouter, again followed by some corrections requiring exchange of the file(s) … and finally they may appear in print and / or online repositories. Taking into account that all participants in the process today are (or should be) well familiar with web-based Content Management Systems and – hopefully – the concept of markup languages, it is simply astonishing that there is no system yet combining the advantages of both. Such a combination could not only serve to shorten the publishing process but also provide the ecosystem for online repositories and web-based collaboration while the results – printable documents – could be updated regularly and made available via book-on-demand and as ePublications. There may be some solutions providing such a system used by publishers “in-house”, but if so, they are not available for free. The paper will propose such a system based on Free and Open Source Software with a simple *proof-of-concept*.

Keywords. LaTeX, Content Management Systems, Free Software,

Electronic publication in the humanities and other fields is still a process following the centuries-old model developed for paper:

1. Authors write papers about research results, including images and tables, using a ‘word processor’ producing a digital document in a proprietary format that hardly can be opened with other software without formatting or information loss.
2. The digital document is sent to the publisher.

3. The publisher (a person working at the publishing house and responsible for this publication) sends the electronic file to someone checking it for errors and guidelines.
4. Before the paper is regarded as ‘finished’, it is sent at least once back to the author for some sort of ‘polishing’.
5. Steps 2 – 4 usually are repeated several times …
6. Finally, the ‘final’ version of the paper is sent to the layouter who transforms the file from the word processor’s format into another usually also commercial and proprietary format, and reworks the entire text according to the layout guidelines.
7. The result may be sent back to the author again for final corrections in a format that he can not change.
8. After the last final ‘final’ reworkings, the file is transformed again into a format suitable for printing and sent to the printer.
9. The printing machine produces the (e.g.) book with binding etc. in the number of the first edition.
10. The printed books are sent to the bookstores or kept in storage until they are ordered.
11. For a few years now, this process is split up after step 7 into two: the second way is the publication in an e-book format.

Except for the usage of digital files sent around several times in several versions via e-mail or some cloud storage, all of this is still identical with the ‘good ol’ paper process’ – and often authors and publishers repeat some of these steps by using prints on paper. So, could this really be called “electronic publishing”? My answer would be: *No!*

In 1991 Tim Berners Lee developed the World Wide Web mostly based on already existing techniques and HTML, to shorten this process. Since the year 2000 many steps of this process could be shortened and united with the help of web-based, Content Management Systems. But 25 and 15 years later, respectively, these systems with databases and versioning are still not used for (printable, well-formatted) publication(s).

The title of my paper is a slight derivation of a quotation from Vint Cerf. Cerf is the (co-) developer of the basic protocol used for the internet, the TCP/IP, and, therefore, one of the ‘fathers of the internet’. Since the early 1970s he took part in crucial developments. Today he is one of Google’s Vice Presidents. Therefore, we should take his warnings regarding the looming ‘Digital Dark Age’ of our data as well as the lack of any solution for the long-time preservation of digital at least seriously. He said these words in February 2015 at the annual meeting of the AAAS. The suggested solution he is working on is the “digital vellum”: It shall provide a soft- and hardware environment able to preserve not

only digital documents but also the software used for their creation and the operating systems as well as the (emulation of) the hardware needed.

Let's put aside the (crucial) questions of (commercial) licences which usually would not allow to run the software in everchanging virtual environments. And let's also put aside other questions about those digital data that are not documents in a broad sense: Most of the databases today may still be able to print out the entries in one set of data, but its form surely will not be sufficient to be regard as a (printed) scientific publication. And, of course, the pure amount of data (sets) will make it impossible to print them out after every change. Let's in addition put aside questions regarding online platforms where data, layout styles and their definitions as well as software for specific functions etc. are distributed over servers all over the world and combined *ad-hoc*. (This should cause a fundamental problem for the "digital vellum".)

So, after letting aside all of these *fundamental* questions and problems: should we follow Vint Cerf's advice anyway and print out our photographs and other documents as long as his "digital vellum" is not available yet? — Of course! Because no-one can guarantee the preservation (let alone: usability) of *any* digital data format for more than 20, let alone 50 years. Archives *plan* to make sure that simple and open formats like TXT and PDF and image formats like TIFF or JPEG will be available for *up to* 50 years, but even this is not sure. File formats like Microsoft Word's .doc are even definitely excluded by archives. Based on experiences with digital data formats from the past 30 years, I surely would doubt any possibility for such a long timespan of preservation: When I started programming in the early 1980s, the answer to the question "How should we store your data for a longest time possible?" would have been: "I put the paper punched tape into a dry and cold armoire." — So, at the moment there is no way to make sure that the digital data we create and work with will be usable even during our own lifetime! Should the creation of such data not be regarded as waste, even willful destruction of life and working time as well as resources? So: Let's print them out, at least those worth surviving the next 50 years!

But with this decision, another problem arises: The layout of our digital documents as they appear on the screen is usually not very satisfactory, often not even acceptable for scientific publishing. This is even more astonishing when we take into account that the basic tools for scientific publication, mostly word processors and type setting programs, have by now been available for more than 25 years. In addition, the tools for web-based content management able to replicate the publishing processes described above also have been available as Content

Management Systems by now for at least 15 years. And both, publishing tools as well as web-based CMS are using Markup Languages …

So: Why is there no unification of both, one widely spread and used as a standard tool and based on Free Software? As far as I know, at the moment there is no such software system available. Some publishing houses use web-based editing tools that their authors may use, but — according to their warnings — even those tools do not generate a document that looks exactly like the one that finally will be printed. And in some of the cases versioning or commenting seems to be difficult. But these tools are 'private', closed source, 'in-house' applications, not available to others. The by far larger group of professional publishing houses does not even offer such tools: They require their authors to follow their specific guidelines, written in MS Word or PDF documents with more than one hundred pages that authors are expected to read first and fully keep in mind! Some publishers offer MS Word templates that can be filled by the authors with their own texts. Only a small number of publishers also offers LaTeX templates. But these are mostly directed to the natural sciences; to humanists and historians, LaTeX and its free interface tools usually are unknown. So, especially these authors are bound (or bind themselves) to non-free word processing software and operating systems and regularly encounter problems should they, for instance, try to reuse or re-work their own documents written some 10–15 years ago with earlier versions of this very same office software.

The solution, from my point of view, would be a combination of a free and open source Content Management System like *Plone* (based on the free Web Application Server ZOPE and its object-oriented database ZODB, both written in Python) and LaTeX: Both, Plone and LaTeX, can be run on any operating system, and a working *proof of concept* solution already exists: It is called *ftw.book*, provided as a Plone module by the Swiss software company *4teamwork*. Because it is free, it could be extended by anyone with some experience in Python and LaTeX into a more general tool, e.g. offering different LaTeX-based designs and document classes or new layouts preferred by the publisher or institution.

What are the advantages of such an extended version of ftw.book?

- Because of the customizable user rights and role management, the entire process described above can be applied to the CMS and adapted to almost any special need of institutions or publishers.
- The CMS versioning allows to go forth and back in the editing process and keep control over the versions at any time.

- No document versions have to be sent multiple times between the participants of the publishing process, because they remain in one place, available to any authorised person.
- The available formatting functions offered in the web interface can be limited to avoid authors breaking them — a big problem in all word processing programs, causing lots of additional work.
- The final print layout is always available for controls.
- This process is protected by the CMS against unauthorised access.
- The final document can be made (un) available over the internet with a few clicks, even if a print version is not yet on or intended for the market. But any authorised reader may print a PDF copy.
- Changes can be easily done while the original is still available.
- Different editions are available at any time, so that links set to an old edition will not break because a new one has been published.
- This would allow, e.g., to update a book on an almost daily basis: When a change has been made to its content, it could not only immediately be online, but also appear in the next printed copy.

While these and other advantages regard the publishing process of scientific documents, there are even more important advantages:

- Not only publishing houses could use such a system, but any institution, group or private person. The software could be used to build up scientific repositories, e. g. for Open Access strategies.
- Because all components of the software scale very well from laptops to (groups of) servers, it would be possible to have a copy of the system run on the personal computers of members of an institution or students, e.g.: They could work on their texts even when they are offline, syncing all preserved versions later while observing the layout required by their institution or publisher.
- A simple syncing tool available for ZOPE guarantees the identity and integrity of the documents on the ‘official’ servers with those on the local computers or laptops.
- With the rapidly growing technical possibilities of handhelds, these should be able to run the entire software system very soon and serve the data to the internet or synchronise them with the server(s). (Any Ubuntu-based tablet already could do this today.)
- The freedom of all components allows for constant development and adaptation of the entire system: from the underlying operating systems to the hardware. Of course, it would be useful to have large research institutions provide central repositories of the freely available parts. Those institutions could even provide

hosting services for projects and, vice versa, require these projects to make their results available online via their repositories in any Open Access strategy suitable.

- Of course: not every paper, article, book etc. would have to be printed: But every one *could* be printed and, therefore, according to Vint Cerf's suggestion, be preserved even for a distant future.

So, everything seems to be wonderful with this suggested solution — or are there disadvantages? Of course, there are some: For instance, if the usability and standard conformity of the system should be preserved, this would radically restrict the many 'bells & whistles' often used in the research projects: Everything that does not fit on a (large) page would have to be excluded. Well, not completely: It would be possible, e.g., to have large images with very high resolutions, annotations, links etc. connected to the reduced images or the data in the printable version. But, of course, such high resolution images, documents or data sets surely will not survive as long as the printed counterpart or 'mother document'.

Another problem could arise from projects where data and information are intrinsically very closely linked to each other. This would make it almost impossible to represent them in a printable form. In these cases a solution could lie in the generation of reduced 'abstracts' or reduced data sets that would be imported automatically from the original database(s) into the suggested system and then be formatted for printing. Again, one would lose some data — but, depending on the decisions made regarding the exported data sets, at least part of the work and resources put into these projects could be preserved for 'eternity'.

The proposed system would not only establish a *real* environment for electronic publishing for the first time, but also provide a solution for the looming dangers of the 'Digital Dark Age' that Vint Cerf and others are warning about and archivists and librarians are or should be aware of. One could even imagine that such system could develop into a general standard for publishing *and* digital preservation. Commercial software then would have to offer plugins to allow its users to publish their texts without having to leave their 'familiar' word processor. For scientific database projects it could offer a solution in the form of repositories that would help to avoid masses of data compiled over years being lost after a short time — just because, e.g., the financial support has been turned off. In cases where server systems spread all over the world are used, e.g. som 'facebook' of science, there should at least be plug-ins to the suggested solution to create printable documents at any time. For such already or soon also very common cases, I do not even see a future in Vint Cerf's "digital vellum".