# Acrobat & the reader: *the design issues*

**Conrad Taylor**

**The ideal paper document and the ideal screen document differ in page orientation, typography, use of colour, and tools for navigation. How can designers ensure that publications released in Acrobat format will work well on both screen and paper? Should alternate Acrobat versions be produced; and if so, what document composition tools make the conversion easier?**

**ADOBE ACROBAT** is a developing technology which, in my opinion, has tremendous potential for publishers. In a previous paper[1] I laid the basis for an understanding of how Acrobat works; in this paper I shall examine the design dilemmas which arise when Acrobat is used as a publishing medium in its own right.

## Messages, media and 'information design'

My perspective on this new publishing medium is that of **information design**. Information designers subjugate the properties of any chosen medium to the task of communicating the message content as clearly as possible. We argue that the *means* of communication must not become a *barrier* to communication, and so champion such values as simplicity in language and maximum legibility in type.

A lot of the skill in information design is invisible to the reader, and this is as it should be. It is a self-effacing discipline, meant to be taken for granted, like the skill of the window-cleaner who removes impediments to your view of the world. Yet every word is chosen with care, and type is selected to suit the viewing conditions. Every linebreak or column break may be assessed to ensure that it will not distract the reader, however subliminally, from the meaning we wish to convey.

A designer needs to understand the conditions under which information will be viewed. Before I design a directional sign, I need to know from what distance and angle it will be seen, how it will be illuminated, and how much clutter there is in the visual environment. I also need to know what visual resources (such as colour) the sign manufacturing process will allow.

**This paper was presented by Conrad Taylor to the conference 'Acrobat in Publishing', organised in London in May 1994 by Merlin Open Systems.**

### CONTENTS IN BRIEF

Or suppose I design a small reference book, such as a travel guide. Because I specify it so, I know that the book will fit in a pocket, the cover is flexible and the pages are 90 gsm matt coated. I know the type will be clear enough to be read at a normal 40 cm distance, that the photographs will be in colour. In fact, I know with certainty how every user will experience this book, except those with visual impairments.

## PDF files and 'viewing instances'

Designing a document for Acrobat distribution is different, because an Acrobat PDF file[2] is not a fixed object for viewing, like a book is.  Your experience of an Acrobat document is always, quite literally, 'mediated'—formed into viewable images on paper, film or computer screen; *those* images are the objects for viewing, comparable to a book.

I shall refer to any one of these viewable output images as a **specific viewing instance** of an Acrobat file.

[1] **Adobe Acrobat: an appraisal**—a fifteen-page paper which I created in support of a presentation to a meeting of the Information Design Association in December 1993. That paper explained Acrobat and the Portable Document Format, made comparisons with other approaches to electronic document distribution, and presented some case studies.

[2] In this paper I am making the assumption that readers are already familiar with Acrobat technology and its Portable Document Format language. If you are unsure of these things, please refer to the papers of other speakers at the *Acrobat in Publishing* conference; or read my earlier paper, *Adobe Acrobat: an appraisal.*

The problem of designing Acrobat-format publications boils down to this: PDF is deliberately, rightly and triumphantly *independent* of the computer platform, the programs used to compose a document, and the fonts used. But when an Acrobat document is viewed or printed, the specific viewing instances which are then generated are *dependent* on the resolution and colour capabilities of the user's monitor or printer.

That simple fact, that one cannot predict characteristics of specific viewing instances, complicates the designer's task. Indeed, the more conscientiously you seek to use the design resources at your disposal to help convey the message, the more frustrated you will be—because you cannot completely predict what those resources will be!

## Display-problematic documents

When you view or print out an Acrobat document, the appearance and therefore the functionality of an Acrobat document may be compromised if your readers do not have computer systems adequate to display all the features of your document.

**Example A: a complex tabloid format newspaper** converted to PDF. Let us further suppose that this newspaper contains many colour photographs, plus detailed maps or information graphics with colour-coding and graduated (ramped) tints.

- ◆ When you open this file with Acrobat Exchange or Reader, you cannot display the whole page and at the same time read the body type on even the largest, highest resolution monitor, so you are constantly scrolling and zooming. If you try to view this publication on a standard VGA 640 × 480 pixel screen, you'll go crazy.
- ◆ To get the best out of the colour photographs, you'll need a 24-bit colour graphics card driving your monitor. Fortunately, Acrobat uses diffusion dithering to give a pretty good representation of 24-bit colour on display systems that represent only 256 colours. But you'll be disappointed if all you have is a 16-greyscale Mac PowerBook 180.
- ◆ The rendering and screen redraw tasks can be considerable, especially for graduated tints and complex vector-mapped graphics; users of slower machines will be drumming their fingers.
- ◆ The large page cannot be printed at the scale at which the designer intended it to be seen.
- ◆ The diagrams have a level of detail in their drawing and captioning which works well in quality litho printing, but not in office colour output.

Obviously, a document designed this way is a poor candidate for electronic distribution via Acrobat.
If there were ever to be a newspaper distributed via Acrobat, it should be reformatted first.

**Example B: a 35 mm slide show** prepared in a program like Aldus Persuasion. It may become common for organisers of conferences to ask speakers to hand over the PostScript output from their slideshows, so they can be converted to PDF and distributed.

In fact this happened after the Seybold San Francisco conference in 1993. A CD-ROM was later sent to all delegates, containing a PDF document over 1,400 pages long, consisting largely of speakers' slides.

Parts of this document demonstrated how *dependent* Acrobat can be for its impact on the system used to view the PDF files. Some of those complex graduated backgrounds take almost 30 seconds to build on screen on a mid-range system, and depend heavily for their effect on the availability of 24-bit colour.

## Which types of document should we redesign for screen viewing?

Considering the foregoing examples, I believe that if a document is likely to be read mostly from computer screens, it makes sense to design it to function best on screen rather than on paper. Publication designers working with Acrobat as a distribution medium should determine the most likely viewing conditions and design their documents accordingly.

**We can usefully divide Acrobat publications into several kinds:**

- ◆ Acrobat documents the design of which *must* be determined by how they will look when printed on paper;
- ◆ Acrobat ephemera, where design is not so critical;
- ◆ Paper-oriented publications additionally converted to Acrobat format, for instance to reach a larger audience over email networks;
- ◆ Publications designed primarily for reading at a computer screen;

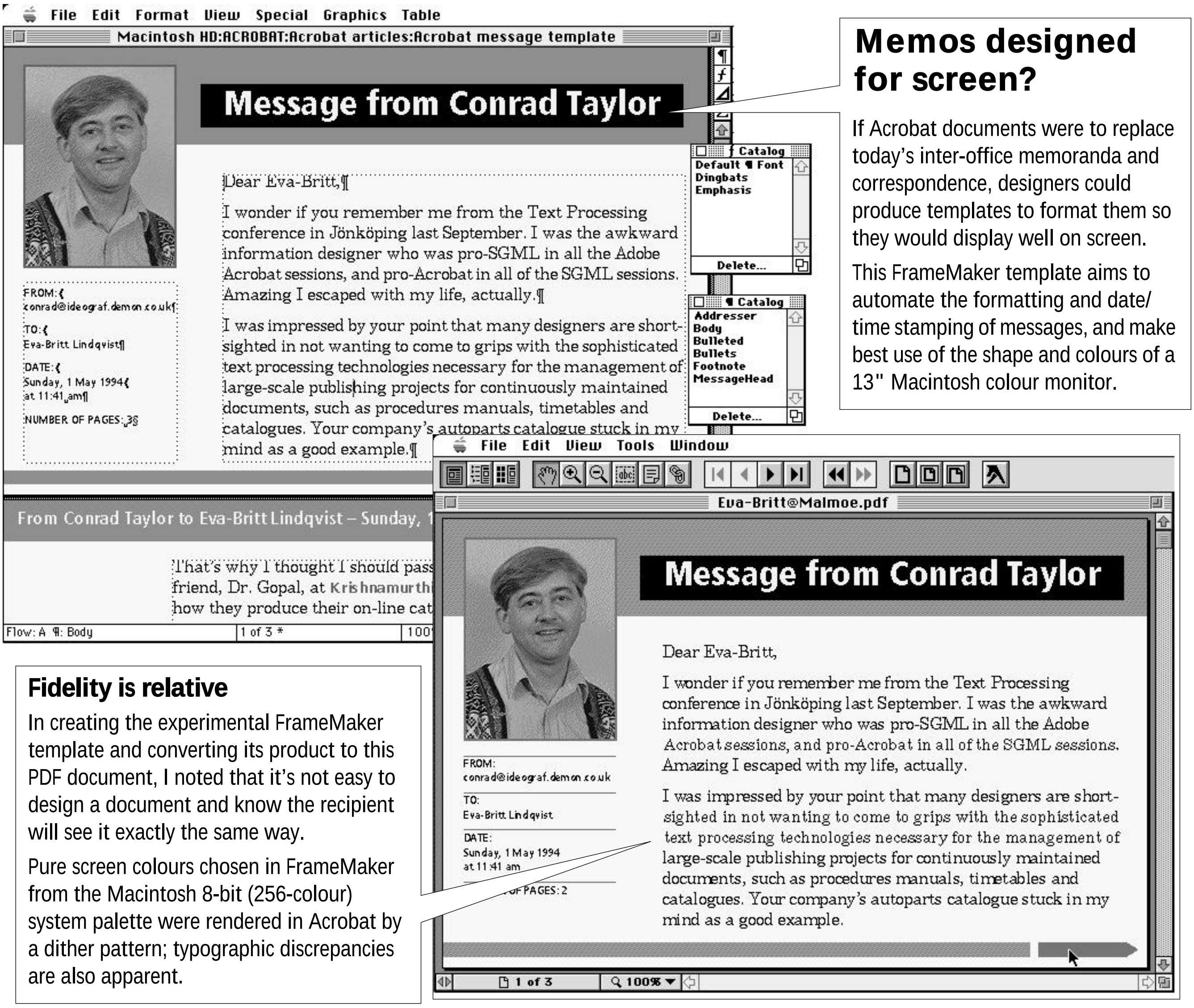
*and, most problematically*:

- ◆ Publications which need to work equally well on screen and on paper.

### When design is determined by paper output

Despite the 'Beyond Paper' slogan, many Acrobat documents are oriented towards print, in which case the document's appearance on screen is of secondary importance, irrelevant to the designer.

- ◆ **Proofing and approvals systems:** when Acrobat is used by a designer to communicate with a client about how a paper publication will look, it is the final appearance of the intended paper document (*e.g.* photoset text, spot colour in graphs, duotone photographs, matt coated paper, spot varnish) which determines the logic of the design. To alter



Message from Conrad Taylor

Dear Eva-Britt,¶

I wonder if you remember me from the Text Processing conference in Jönköping last September. I was the awkward information designer who was pro-SGML in all the Adobe Acrobat sessions, and pro-Acrobat in all of the SGML sessions. Amazing I escaped with my life, actually.¶

I was impressed by your point that many designers are short-sighted in not wanting to come to grips with the sophisticated text processing technologies necessary for the management of large-scale publishing projects for continuously maintained documents, such as procedures manuals, timetables and catalogues. Your company's autoparts catalogue stuck in my mind as a good example.¶

FROM:{
conrad@ideograf.demon.co.uk{
TO:{
Eva-Britt Lindqvist¶
DATE:{
Sunday, 1 May 1994{
at 11:41 am¶
NUMBER OF PAGES: 3§

From Conrad Taylor to Eva-Britt Lindqvist – Sunday, 1

That's why I thought I should pass
friend, Dr. Gopal, at Krishnamurthi
how they produce their on-line cat

Message from Conrad Taylor

Dear Eva-Britt,

I wonder if you remember me from the Text Processing conference in Jönköping last September. I was the awkward information designer who was pro-SGML in all the Adobe Acrobat sessions, and pro-Acrobat in all of the SGML sessions. Amazing I escaped with my life, actually.

I was impressed by your point that many designers are short-sighted in not wanting to come to grips with the sophisticated text processing technologies necessary for the management of large-scale publishing projects for continuously maintained documents, such as procedures manuals, timetables and catalogues. Your company's autoparts catalogue stuck in my mind as a good example.

FROM:
conrad@ideograf.demon.co.uk
TO:
Eva-Britt Lindqvist
DATE:
Sunday, 1 May 1994
at 11:41 am
OF PAGES: 2

### Memos designed for screen?

If Acrobat documents were to replace today's inter-office memoranda and correspondence, designers could produce templates to format them so they would display well on screen.

This FrameMaker template aims to automate the formatting and date/time stamping of messages, and make best use of the shape and colours of a 13" Macintosh colour monitor.

### Fidelity is relative

In creating the experimental FrameMaker template and converting its product to this PDF document, I noted that it's not easy to design a document and know the recipient will see it exactly the same way.

Pure screen colours chosen in FrameMaker from the Macintosh 8-bit (256-colour) system palette were rendered in Acrobat by a dither pattern; typographic discrepancies are also apparent.

the design in any way would invalidate Acrobat's use as a proofing system. Also, as the communication circuit in a proofing system is limited to a few people, it is easy to make sure that everyone has a 24-bit colour system to see the document as it will appear in print. Minus spot varnish, of course.

◆ **Document archiving:** Acrobat may be used to store an electronic version of a paper document: a better solution than microfilm or document image storage systems. Again, fidelity to the original printed appearance is the over-riding consideration.

◆ **Distributed printing:** In these applications, Acrobat is merely a convenient way of getting a document intact onto somebody's system so they can print it.

*Example*: MetLife's forms design department distributes all the company's internal forms as Acrobat files on CD-ROM, and local offices print them out as they need them.

*Example*: the U.S. tax forms are on Compuserve in Acrobat format.

Normally, people using Acrobat to support distributed printing will design documents in a 'lowest common denominator printing' format, *e.g.*, to look good from a 300 dpi laser printer. But Acrobat's independence can still be compromised by the different proportions of the ISO and American standard paper formats (A4 *vs.* US Letter).

(*Note*: many PostScript output bureaux had hoped Acrobat could serve as a way of transferring files from customers for filmsetting. Apparently this doesn't work at all well yet.)

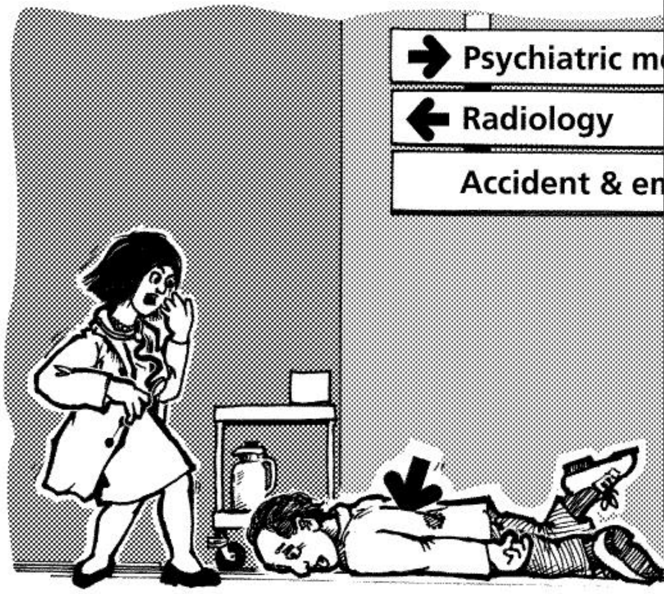
### 'Acrobat ephemera': worth design effort?

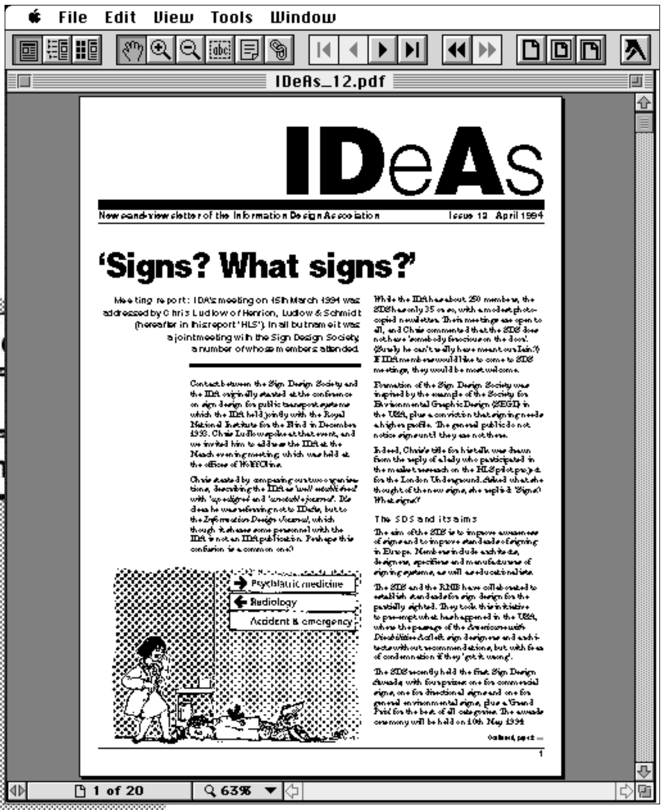
By this term I mean internal office documents such as memoranda, minutes, proposals, notices, reports and agendas.

Adobe's little book *Beyond Paper*, by Patrick Ames, expresses the hope that such documents could circulate purely in Acrobat form within an organization's computer network, never going near a laser printer, office copier or fax machine and so saving hundreds of thousands of tons of paper.

## IDeAs: optimized for A4 print

Although the Information Design Association's newsletter is being rendered into PDF for electronic distribution, it is not designed for the screen, and indeed only looks its best printed on A4 paper. The cartoon is rendered as a 300 dpi bitmap—ideal for laserprint reproduction, but it scales badly, producing moiré at other ratios, and does not compress well in the PDF file.





IDeAs

News-and-views letter of the Information Design Association Issue 12 April 1994

'Signs? What signs?'

Meeting report: IDA's meeting on 15th March 1994 was addressed by Chris Ludlow of Henrion, Ludlow & Schmidt (hereafter in this report 'HLS'). In all but name it was a joint meeting with the Sign Design Society, a number of whose members attended.

Contact between the Sign Design Society and the IDA originally started at the conference on sign design for public transport systems which the IDA held jointly with the Royal National Institute for the Blind in December 1993. Chris Ludlow spoke at that event, and we invited him to address the IDA at the March evening meeting, which was held at the offices of WolffOlins.

Chris started by comparing our two organisations, describing the IDA as 'well established' with 'a pedigree' and 'a reputable journal'. I'm sure he was referring not to IDeAs, but to the *Information Design Journal*, which though it shares some personnel with the IDA is not an IDA publication. Perhaps this confusion is a common one?

While the IDA has about 250 members, the SDS has only 35 or so, with a modest photocopied newsletter. Their meetings are open to all, and Chris commented that the SDS does not have 'somebody ferocious on the door'. (Surely he can't really have meant our Iain?) If IDA members would like to come to SDS meetings, they would be most welcome.

Formation of the Sign Design Society was inspired by the example of the Society for Environmental Graphic Design (SEGD) in the USA, plus a conviction that signing needs a higher profile. The general public do not notice signs until they are not there.

Indeed, Chris's title for his talk was drawn from the reply of a lady who participated in the market research on the HLS pilot project for the London Underground. Asked what she thought of the new signs, she replied: 'Signs? What signs?'

The SDS and its aims

The aim of the SDS is to improve awareness of signs and to improve standards of signing in Europe. Members include architects, designers, specifiers and manufacturers of signing systems, as well as educationalists.

The SDS and the RNIB have collaborated to establish standards for sign design for the partially sighted. They took this initiative to pre-empt what has happened in the USA, where the passage of the Americans with *Disabilities Act* left sign designers and architects without recommendations, but with fear of condemnation if they 'got it wrong'.

The SDS recently held the first Sign Design Awards, with four prizes: one for commercial signs, one for directional signs and one for general environmental signs, plus a 'Grand Prix' for the best of all categories. The awards ceremony will be held on 10th May 1994.

Continued, page 2 —

1

The paper versions of these short-life office documents are mostly only a few pages long, and therefore simple to navigate. Such documents, especially if they have the character of personal communication, rarely warrant an expert designer's intervention, and in most organizations little attempt is made to control their appearance.

**Does Acrobat technology change this?** Perhaps ideally it should, just as documents designed to be faxed should look different from documents sent by post.[3]

Design intervention in this area could come in the form of standard template documents for making Acrobat message files, issued to staff for use with their word processing software. I give an example at the top of the previous page, but somehow I doubt it will happen; even if it did I doubt that the sum of human happiness would greatly increase thereby.

### Paper documents converted to PDF to facilitate network distribution

Some Acrobat publications will start out with their design determined by the paper-printed version; their conversion to PDF will at least initially be simply to archive them and facilitate their transmission over computer networks.

However, if the readership for the PDF-transmitted version of such a publication becomes a significant total of the entire readership, it would be logical for the publication's designers to consider this and redesign accordingly.

For example, take **IDeAs**, the bimonthly newsletter of the Information Design Association, which I currently edit and produce. This low-budget publication is litho printed from 300 dpi laserprint originals, with black ink on uncoated paper, 16–20 pages wire-stitched, to a finished size of A4. There are no photographs, but black-and-white cartoons are included as 300 dpi TIFF files, with a single bit per pixel; and black-and-white screen dumps are processed to the same specification.

Thus the design of IDeAs, and the methods used to process graphics, are dictated to by the limitations of the printing budget.

After completing an edition and sending laserprint originals for litho printing, I use Acrobat Distiller to make a PDF version of IDeAs. We are considering putting this on the Internet somewhere at an **ftp site**.[4]

[3] I maintain different template documents for writing letters and preparing fax transmisions. Documents for faxing have an integral fax header on the first page, and use larger type with slightly loose letterspace. Similarly I could create a template for Acrobat-format 'letters', as in the experiment illustrated on page 3.

[4] **Ftp site** = a computer with a permanent connection to the Internet which maintains an archive of documents and software, for downloading using the **file transfer protocol.** Many university computer departments are ftp sites.

However, we have no intention of modifying the design to optimize IDeAs for Acrobat readers.

The current Acrobat version of IDeAs differs in design from an 'ideal' version. For instance, the tint patterns I use in cartoons for IDeAs are fixed 300 dpi dither patterns, and therefore they scale badly. US readers printing a copy of IDeAs onto 8½" × 11" 'US Letter' paper will have to use Acrobat's 'Reduce to Fit' printing option, which will mash up these dither patterns.

### Acrobat documents designed for screen

The final type of Acrobat document to consider is one which from the start has been designed for viewing at a computer monitor. Indeed Acrobat's 'added value' electronic-document features such as Links and Bookmarks are good reasons why publishers should consider an Acrobat publication as potentially a very different kind of animal from a printed document.

Ignoring for the moment these hypertext-like elements and concentrating simply on how a publication looks to its reader, it is clear that pages and computer monitors are different media. I now want to consider how a designer should think about a screen document, and how it should differ from a page-printed document.

(*Later, I shall consider the dilemma that is faced when a document may be required in both a screen version and a printed version.*)

## Screens versus pages: display area, ratio & orientation

Printed pages are generally 'portrait', and with few exceptions computer screens are 'landscape'.[5] Screen displays of information are therefore more sensibly designed to be wider than they are tall.

The ratio of an A4 page is 1:√2, approximately 1:1.4. The ratio of a US Letter page is approximately 1:1.3.

Computer screens come in different sizes, with different numbers of pixels, but the screen ratio is remarkably constant. Some pixel counts for graphics screens are:

| | | |
|---|---|---|
| ◆ | 640 × 480 | (VGA Graphics, Macintosh 13", Mac PowerBook 180c) |
| ◆ | 800 × 600 | (SVGA) |
| ◆ | 832 × 624 | (Macintosh 16") |
| ◆ | 1024 × 768 | (SVGA, IBM 8514A) |
| ◆ | 1152 × 870 | (Macintosh 21") |

All these have the same ratio of 3:4, or 1:1.33.

Some other formats do exist which have different ratios. Non-colour Mac PowerBooks have 640 × 400 pixels, the same width as the 13" monitor but not so high, giving a ratio of 1:1.6; the Apple Portrait Display is 640 × 870 pixels, a ratio of 1:0.74 or 1.36:1.

### Usable Acrobat display area

However, designers must remember that not every pixel on a screen is available for displaying an Acrobat document. Even when an Acrobat window is expanded to 'fill' the screen, pixels are stolen from the edges:

- **The computer system's menu bar**: on a Macintosh this is 19 pixels high;
- **the Acrobat toolbar**: this is 32 pixels high;
- **the Acrobat document window title bar:** this is 18 pixels high;
- **the Acrobat document window bottom display/ scroll bar:** this is 17 pixels including its 'shadow'.
- **the vertical scrollbar**, which is 17 pixels wide.

In addition there are extra pixels which Acrobat arranges by default around a document when it opens into a window. All in all, I estimate that we should discount for any given display 89 of its vertical pixels, and 25 of its horizontal pixels. Thus on a 640 × 480 display, the space left to view an Acrobat document is actually 615 × 391 pixels.

### Ratio of usable display area

So in fact, a sensible ratio for designing an Acrobat document is *not* 3:4 (1:1.33) like a computer screen, but something else which takes into account what is subtracted from the edges.

In my example shown at the top of page 3, I actually designed my page to be 615 × 391 points, since in graphical displays a point, defined as 1/72", is mapped to a pixel on the screen. This page ratio is 1:1.57, narrower than either a landscape A4 or US Letter page.

The more pixels on a screen, the less relatively speaking will be consumed by menu bars, toolbars and the like. Thus on a Macintosh 21" screen the usable display area for an Acrobat document is 1127 × 781 pixels, a display ratio of 1:1.44, almost exactly the same as A4. But since we do not know what kind of monitor the reader will have, designing for the most common size (640 × 480) makes most sense.

### Caveat: space for bookmarks

The publisher of an Acrobat document may consider that it is best navigated by the use of **Bookmarks**, which display in a subsidiary scrolling panel that claims a slice off the left side of the Acrobat document window.

In consequence, the remaining space for the display of an Acrobat document itself will be squarer. Might one then best create an Acrobat screen-oriented document to measure, say, 430 × 391 points?

[5] The exceptions include 'portrait' displays, and monitors like the Radius Pivot series which may be used either portrait or landscape.

## Screens versus pages: information resolution

What is the resolution of a printed page? How long is a piece of string? Are these similar kinds of question? Not quite, but in both cases the answer starts, *It depends.* In comparison, computer monitors have predictable pixel resolutions in the range 70–90 pixels per linear inch. The Macintosh display standard, for instance, is 72 pixels per inch, or 5,184 per square inch.

The 'resolution' of a paper document is governed largely by the characteristics of the equipment used to mark it, but is further complicated because the continuous-tone elements on the page, such as areas of tint and photographs, are rendered by most printing processes using a halftoning method.

**It should be noted that I am using the term 'resolution' from a reader's perspective,** *i.e.*, with what detail does information exist on a page?

### Example 1: comparing a monitor to a filmset, litho-printed book

Take, for example, a book printed on coated paper, with a screen frequency for photographs and tints of 150 lines per inch (60 lines/cm). Let us further suppose that the filmsetting for this book was done at 2,540 dpi (1,000 dots/cm). How does this compare with the resolution of a 72 pixels-per-inch computer monitor?

- **Type:** This is rendered at the full resolution of the imagesetter. We can say that the type has more than **35 times** the resolution of the screen.
- **Scanned line images:** these may have up to the same resolution as the type—*if* each scanned pixel is imaged by one imagesetter pixel. However, a line drawing scanned at 500 dpi and set to film same-size at 2,540 dpi will have each of its scanned pixels imaged by a block of 5 × 5 = 25 imagesetter pixels. In effect, that image is only seven times sharper than you'd see it on computer screen.
- **Photographs:** the effective resolution of a photograph is determined by its halftone screen ruling. No detail smaller than a halftone dot can be transmitted. A screen ruling of 150 lines per inch is only **twice** as high as that of a typical greyscale computer screen — which, it must be remembered, is a continuous-tone output device.
- **Vector art with tints:** Maps and diagrams made with programs like CorelDRAW! or Illustrator have mixed information resolutions when litho printed. The solid line elements are imaged at the same resolution as the type, but tinted elements have the same effective visual resolution as the photographs.

  (In comparison, the same images on a greyscale screen in an Acrobat document would have tints rendered as continuous tone; line and tone areas would have the same informational resolution.)

### Example 2: comparing a monitor to a 300 dpi laserprint document

What I have said about different page elements in a filmset book also applies to a laserprinted document, but the figures are different. At 300 dpi, type and line art is four times sharper than a screen version would be; at a laser printer's default halftone screen ruling of 53 lines/inch, it is the screen version which packs more image information per square inch!

### Example 3: comparing a monitor to a 300 dpi dye-sublimation print-out

Dye-sublimation prints are continuous-tone in the sense that although they have a fixed pixel resolution, each pixel may be imaged at one of many thousands of colours. A 300 dpi dye-sublimation print will therefore have four times the resolution of the computer monitor both for type *and* for pictures. Indeed, the pictures will have twice the information resolution of a book with pictures screened at 150 lines per inch!

There is a further twist: some of the dye-sublimation printers[6] enhance the *apparent* sharpness of type and vector-art line work by **anti-aliasing** techniques. That is, if a pixel is partially clipped by the outline of a type character, that pixel is imaged to a partial density. It could be argued therefore that the informational resolution of such type exceeds the nominal resolution of the printer's marking engine.

(Anti-aliasing could in theory be used to increase the apparent resolution of type on computers, as is done for TV titling and subtitling; but current standard computer operating systems and graphic subsystems do not support it. In any case, at the low resolution of computer monitors, many people consider that anti-aliased type looks too fuzzy and out of focus.)

## Comparing screens: colour and 'bit depth'

We saw on page 5 how computer display systems may vary in the number of pixels they show. They also vary in their ability to display a range of colours.

This may be a limitation of the screen—a monochrome screen will never show colour, and the LCD screen of the PowerBook 180 I am using to write this can show only 16 greyscales.

Or the limitation may be in the circuitry driving the screen: the Sony 16" monitor I am driving from my PowerBook shows 256 colours, the maximum that can be encoded in a single byte of storage ('8-bit colour'); but on a Centris 660 it shows thousands of colours, encoded using two bytes ('16-bit colour'), and on a Power Mac 8100⁄80 it boasts a palette of 16,777,216 colours, encoded with three bytes ('24-bit colour').

[6] Example: Tektronics Phaser IISD.

## Pure and dithered colours

How does Acrobat cope, not knowing what kind of screen a user will have available to display a document? Remarkably well, in fact. A 24-bit colour image in an Acrobat document is represented on an 8-bit display using a technique called **dithering**.

On an 8-bit colour display, a single pixel may be any one of 256 colours in the computer's '**system palette**'. To approximate the appearance of a colour not in this palette, Acrobat divides the image up into multiple-pixel blocks, and within each block creates a pattern of pixels set to different colours available from the 256. At a normal viewing distance, the eye is supposed to integrate this pattern, to see not the pattern but the illusion of continous colour.

This method of representing colour is analogous to halftoning in print, but with 256 colours to play with instead of just cyan, magenta, yellow and black. Another difference is that in normal colour printing, the dots of coloured ink vary in size, whereas pixels on a screen are all the same size. To give the smoothest appearance of colour, Acrobat uses an apparently irregular or pseudo-random pattern of dithering, known as **diffusion dither**.[7]

## Implications for information design

At normal viewing distances, the granularity of the dither pattern on an 8-bit computer monitor is pretty obvious. Designers must bear this in mind, because 24-bit displays are not commonplace. The Acrobat documents we produce must work well on 8-bit displays, dithers and all.

The problem as I see it is not so much with photographs, although the loss of image definition caused by dithering is regrettable. Rather, it is the definition of fine coloured type, coloured fills and line work which suffers most from diffusion dithering—and which has the most to gain if we can work in pure screen colours.

A print analogy is again useful. Consider printed maps, such as those made by the Ordnance Survey. They are not simple four-colour print jobs. The brown contour lines and the green symbols for trees are printed in brown and green ink, not simulated with tints of cyan, yellow, magenta and black. If tint simulations were used, the contours and tree symbols would lack the edge definition that is gained by printing solid lines in special colours.

Similarly, if we are restricted to four-colour print, the edge definition of fine type is successful if printed in a red created by solid yellow surprinted on solid magenta, but unsuccessful in a colour simulated by a 40% tint of cyan atop a 30% tint of magenta.

## Choosing and using indexed colour

If you have an 8-bit colour display, you see only 256 colour as pure colours. But *which* 256 colours? This varies from one computer system to another.

I have investigated Macintosh 8-bit colourspace, and find it to be referenced to 24-bit RGB colourspace. Each of the 256 colours in the system palette is defined with a red, a green and a blue value.

Although in 24-bit colourspace there are 256 possible levels for the strength of *each* of these three colours, within the Macintosh 8-bit colourspace only a selected number of these colours are used.

It is useful for a designer to know RGB co-ordinates of these system colours, because if you use these numbers to define colours for lines, type and fills in Acrobat documents, they should appear as pure and undithered on 8-bit Mac systems as on 16-bit or 24-bit systems. For reference, the Appendix to this document shows samples of all these colours and their RGB values.[8]

(Remember that these are Macintosh system colours. VGA systems appear to be less predictable, and some VGA systems can display only 16 colours.)

## Zone of mystery: controlling [ /Indexed /DeviceRGB ]

For the graphic designer, one of the most frustrating aspects of Acrobat is how little control one gets. It's like using a sophisticated instant camera designed for the amateur: if you want to do more than point-and-click, you have to learn how to write in assembler code and blow new EPROM chips for the camera!

(The equivalent, in the case of Acrobat: learn how to embed commands in the PostScript code fed to Distiller, or rewrite the **setdistillerparams** code.)

At the time of writing I am still unsure how to force Acrobat to use the system palette in an 8-bit colour system to display colours which have been defined in the first place with reference to that system.

While creating the experimental 'Acrobat letter' design shown on page 3, I found that although I limited myself to system colours for all my type, line and colour fill work, these were dithered when viewed in versions of PDF created with Acrobat Distiller. However, they displayed in pure indexed colour in PDF documents created with PDFWriter. Investigating the two PDF files with a text processor, I discovered PDFWriter had included the definition `[ /Indexed /DeviceRGB]`, which appears to have forced Acrobat to use indexed colour. But how do we, as non-technical users, control this parameter for a Distilled document?

[7] This method of representing continous-tone images by dots of equal size spaced unequally is now also being used in colour printing, where it is known as 'stochastic screening' or 'frequency-modulated (FM) screening'.

[8] The Appendix is not included in paper versions of this document, as the budget does not stretch to colour printing, but I am trying to prepare it as a separate PDF document. The conversion to PDF is proving problematic, so I cannot guarantee the fidelity with which these colour samples are displayed, but I shall speak for the accuracy of the colour co-ordinates.

# Designing for screen, designing for paper

**NOW WE HAVE CONSIDERED** the technical differences between screen and paper display, we can move on to establish some broad guidelines about what works best for screen-oriented Acrobat documents in comparison to their paper-oriented cousins.

### 'What the butler saw'

The page as a design unit is compromised if we force our readers to view it as if through a keyhole. As I argued on page 2, a newspaper page works as a design unit because when you hold it in your hands you can see all of it at a glance and still read the smallest type, whereas if it were rendered without reformatting into PDF you could not simultaneously view the whole page and read the type. I think this is wrong.

## Page format

**Pages for screen should be designed landscape rather than portrait**. On the majority of monitors, which have 640 × 480 pixels, Acrobat will leave a display area of about 615 × 390 pixels for a document, and this can guide you in your page set-up.

However, if Bookmarks are to be a major tool for page navigation, space needs to be left for the Bookmarks display panel, and so the page will be squarer, closer in proportion to 430 × 390 pixels.

**Margins**: a 'page' displayed on screen already benefits from 'margins' provided by a document window. If the document will never be printed, narrower margins than usual may work. If the document may be printed, at least 1 cm margin should be left to accommodate the printing limitations of most page printers.

## Typography

Since print resolution is 4–30 times higher than screen resolution, as explained on page 6, typography for screen must be different from typography for paper:

### Body text

- ◆ Type for screen should be larger than for print. Ten or eleven points is an absolute minimum for most designs of type, but some will need to be even larger. This Plantin type is set at 10.3 pts, adequate for printing to paper in a narrow column but not easy to read at 100% from screen.
- ◆ Clear, uncluttered type designs are best for display on screen. Use plain sans-serif faces, or else faces with firm serifs. Beware faces which have too much contrast either of stem weight (*e.g.* Bodoni, Times Bold) or of set width (*e.g.* Avant Garde).
- ◆ With larger type and a smaller page display, single-column setting makes more sense. But the column of type may still be too wide for easy reading if it is set the whole width of the display. The remaining lateral space may therefore be used for headings or navigational aids such as hypertext buttons.

### Section headings, subheadings

One useful function for headings in a document is to give readers a 'fast track' so that they can skim through topics to find what they want. For an on-screen document, which has less text at a showing than a print document, the relationship between headings and navigational aids such as running heads (*q.v.* below) may need to be rethought.

- ◆ Subheadings in a column of type normally draw attention to themselves by being bolder and slightly larger. These signals may be too subtle for on-screen documents. Positional clues (*e.g.* negative indent) and bright spot colour may need to be used too.
- ◆ Because of the landscape orientation of a screen-oriented page, **sideheadings** may be more sensibly used than in-line (column-embedded) headings. This may have implications for your choice of page make-up software; not all DTP programs handle sideheadings gracefully.[9]
- ◆ The '**Bookmarks**' panel of an Acrobat document may serve the same purpose as either the Table of Contents or the Index of a conventional paper document—or both. Use bookmarks to help readers find their way to the level of heading they require. Consider alternative methods of presenting topics in the bookmarks panel, for instance alphabetical as opposed to sequential. Or both!

### Running heads, &c.

You cannot 'flip the pages' in an on-screen-book, and only by looking at the status of the vertical scrollbar and page-number display do you get much sense of where you are. However, you could help by generating more informative running heads:

- ◆ Concatenate document levels in your running heads: *e.g.* section heading, major heading and minor heading: '**PROPOSALS: Environmental improvements: *footpaths***'. This gives readers a better sense of where they are in a document. Use typography or colour to distinguish the levels within the running heads.[10]

[9] For instance, PageMaker and QuarkXPress cannot handle sideheadings except as separate text blocks. Ventura Publisher has always been able to, and FrameMaker can from versions 4.0 and up.

[10] Again, this requires the use of a document composition program with a sophisticated approach to the automatic creation of running heads. Ventura Publisher and FrameMaker are both recommended.

## Colours

Putting colour on a printed page is expensive. In contrast, with Acrobat you can 'publish' a full-colour screen-oriented document inexpensively.

- **Background colour:** for purely screen-based documents, the whole 'page' may be given a background colour. You could even set a very dark page, with reversed type, like a slide show.
- **Line work in CMYK colour:** to get predictable colour and sharp type and line work in a printed document, you need to design using CMYK colour. But this may not show well on screen, due to dithering.
- **Line work in indexed colour:** Flat, pure indexed RGB colour is best for 8-bit colour displays, but unpredictable for printed versions. Colour boundaries which display sharply on screen in indexed colour may print out indistinct and fuzzy if the CMYK tint-simulation of one colour butts up against that of another colour.
- **Colour and contrast in line work:** when colour lies next to colour, for instance type on a background or line work/fills in a diagram, ensure that you create sufficient contrast in the **luminosity** of the colours. That way your type and line work will remain visible on a greyscale or 16-colour system.

## Pictures

Decisions which designers make about pictures will affect not only their screen appearance, but also how well they will print, and how large the resulting PDF document will end up in units of memory.

- **Scanned image resolution:** for photographs, the recommendation is that the pixel resolution of a continuous-tone image (colour or greyscale) should be at least 1.5× and preferrably 2× the halftone cell ruling of the printing device it will be sent to.[11] For a 300 dpi printer (*e.g.* laser printer or colour wax transfer printer), the halftone screen ruling is typically about 60 lines/inch, therefore a scanned image pixel resolution of 120 dpi is best.

  Therefore, image print quality may be compromised if we allow Acrobat Distiller to subsample images as low as the default 72 dpi resolution.

  Another factor to bear in mind is that Acrobat has the ability to zoom in to retrieve higher-resolution image data than can be seen at 100%. For images where detail provides important information (X-rays?), very high resolutions may be required, and a Link may be used to zoom the view to highlight the important detail.
- **JPEG Compression?** This 'lossy' compression scheme for continuous-tone images may compromise viewing quality and print quality; but if detail is important as in the cases just outlined, JPEG compression may be the only way to contain file size. The designer must assess the trade-off of costs and benefits.
- **Avoid dithered halftones (unless)…** As described on page 4, dither patterns can give excellent results on a printer of known resolution, but produce moiré at other resolutions and do not compress well through CCITT compression. They are not recommended for documents intended primarily for screen viewing.
- **Vector images: a reader may zoom them.** Map or diagram detail too small to print may be acceptable in a screen document if the image was created with a vector-based drawing program such as Adobe Illustrator or CorelDRAW! This is because the reader can zoom in on necessary detail, guided perhaps by Links.

  Alternatively, the diagram may have to be designed to make sense at different levels, at different magnifications. This takes diagram design into new areas, and the design may not function well in print, requiring alternate print and screen versions.
- **Consider 'screen build'.** Acrobat documents have to be interpreted to screen, and complex vector art takes a long time to render to screen.[12] Avoid graduated fills to boost screen performance.

## Navigation

The provision of 'hot links' between pages and views in an Acrobat document, and how to represent them visually on screen, is one area where we can expect design ingenuity to play a great role in making screen documents more accessible. Some useful gadgets:

- **Constant buttons** for 'home', 'Index' or whatever. Art for these should be set up on a master page. (At present, software trickery is required to ensure that links are properly inherited on every page.)
- **Intelligent fast-forward buttons** which take the reader to the next significant section, *e.g.* start of next chapter.
- **Context displays** which show what topics come next and which precede; clicking on a topic name should take you to that topic.

As more vendors of document-composition software directly support the **pdfmark** operator in the Portable Document Format language, such navigation aids should become easier to set up and revise.

[11] Non-halftoning, continuous-tone printers such as dye-sublimation printers ideally require one scanned image pixel to drive each pixel on the printer.

[12] See my discussion of this in *Adobe Acrobat: an appraisal*, pp 10–11.

# Making it easy with the right design tools

**A DESIGNER CANNOT AFFORD** to ignore the production environment, so finally I want to consider which software tools are most appropriate for the authoring and design of Acrobat documents.

One of the strengths of Acrobat is that any program which can produce PostScript can be the environment for designing an Acrobat document. But I strongly believe that some programs are better suited than others, especially when a document must be produced both in an ideal paper-oriented version and an ideal screen-oriented Acrobat version.

## The need for 'generic mark-up'

Whenever a document design needs to be transformed for different display and output circumstances, we should avoid using authoring systems which format documents with **procedural mark-up**. That term is 'text-processing-speak' for the use of inserted codes, whether visible or hidden, to define the appearance of output directly.

Instead, we need to move towards authoring systems based on **generic mark-up**. That means systems which use inserted codes, whether visible or hidden, which describe chunks of text, graphics and other document components as being a certain kind of *entity*, or having a certain kind of *function* in a document.

At a separate level, such entities are associated by the designer with a particular *appearance* and *behaviour*, so we may decide that all level-2 subheadings shall be set in Frutiger and show as green, that all chapter headings shall start a new page, that all pages using the page-tag 'reference' shall have a yellow background, *etc*.

### The case for (and against) SGML

An extreme case of a generic mark-up system is the **Standard Generalized Mark-up Language** or **SGML**, which is defined in an ISO standard (ISO 8879). SGML mark-up says nothing at all about document appearance, and rendering the mark-up into a formatted document is considered by SGML devotees to be a completely separate step, often known as the **Formatting of a Specific Instance (FOSI)**.

I strongly believe in SGML and admire those organizations heroic enough to tackle the arcane and unfriendly process of using SGML tools in publishing. The Oxford English Dictionary and the legal encyclopedias of Her Majesty's Stationery Office, for example, have benefitted from this approach. But it is too complex a route for many publishers to travel. SGML publishing tools are still not accessible enough.

### Document composition software with substantial generic mark-up

Over the years, DTP programs have adopted more generic forms of mark-up, known to many designers as **DTP stylesheets**. For instance, in Aldus PageMaker, typographic styles are defined and associated with each type of paragraph. If the definition of the typographic style for 'Subheads' is later revised, all paragraphs which have been assigned that style are reformatted.

But the commonest DTP programs, QuarkXPress and PageMaker, implement the power of generic mark-up in only a superficial way. They therefore make it difficult to transform a document efficiently between formats suitable for such radically different forms of display as Acrobat publishing may demand.

Instead, Acrobat publishers should be looking at more structured publishing tools such as **Ventura Publisher** and **FrameMaker**. These programs implement generic mark-up at a deeper level, and allow publishers to exchange one set of format definitions for another.

By associating an alternative Stylesheet with a Ventura Chapter file, for instance, a Ventura document may change page sizes and colours, be dressed in different typography, and substitute side-headings for in-line headings. One can therefore have both a page-oriented stylesheet and a screen-oriented stylesheet, and swap to the latter before generating PostScript to 'distill' into an Acrobat document.

## The benefits of using FrameMaker

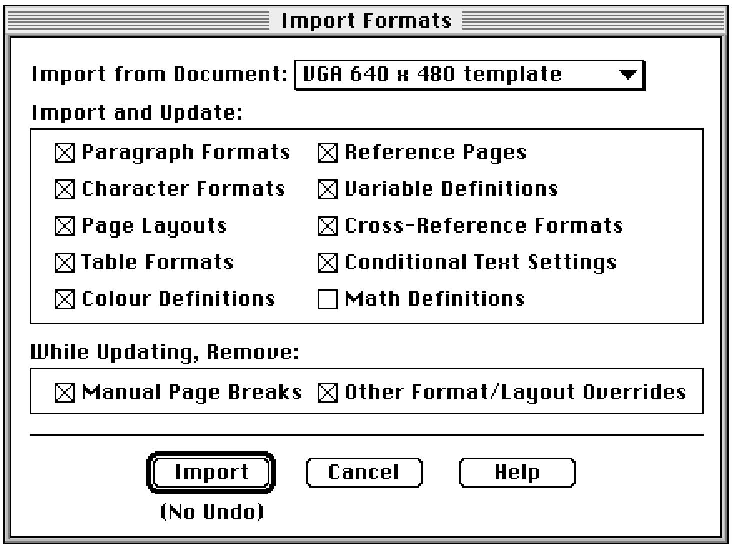
My top choice of document processor for this job would be **FrameMaker 4.0.2**. [13] Formatting in FrameMaker can be even more rigorously generic than in Ventura, since typographic styles can be set up not only for paragraphs, but also for exceptional character styles within paragraphs.

To implement a multi-format publishing programme using FrameMaker, two separate template documents should be developed—for instance, one for paper, and one for screen display. Setting up a template document can require a lot of thought, as you define a number of master page layouts, paragraph styles, character styles, table styles, ruling styles, variables, colour definitions, categories of conditional text and so on.

It takes even more discipline to develop alternate templates in parallel, because the names for all these entities must be identical across both templates.

One of these templates is then used to author the document. As text is imported or authored directly in the Frame document, paragraph and character styles are applied. Graphic frames are created and pictures

[13] FrameMaker 4.0.2 is a recent upgrade to FrameMaker, first being made available to Mac users, which offers direct support for pdfmark in a similar manner to Ventura. It is also faster than FrameMaker 4.0



Import Formats

Import from Document: VGA 640 x 480 template

Import and Update:

☒ Paragraph Formats
☒ Character Formats
☒ Page Layouts
☒ Table Formats
☒ Colour Definitions
☒ Reference Pages
☒ Variable Definitions
☒ Cross-Reference Formats
☒ Conditional Text Settings
☐ Math Definitions

While Updating, Remove:

☒ Manual Page Breaks
☒ Other Format/Layout Overrides

## FrameMaker: global reformatting using an alternate template

The **Import Formats** command allows you to overwrite the current definitions of many style formats, page layouts, variables definitions etc., by bringing in alternative definitions from another template document.

If you have been rigorous in applying all formatting through these entities, and the equivalent entities in both templates are named identically, the document will be automatically and comprehensively reformatted to the specifications contained in the alternate template.

are imported ‘by reference’ into them. Tables may be created using the pre-prepared table styles; cross-references embedded; and so on.

When one edition is complete, a copy is made and the alternative template is then also opened. Calling up Frame’s **Import Formats…** dialogue box lets you nominate the alternative template as a source for alternative style definitions (*see above*). If you have been rigorous in matching styles and entities on a one-to-one basis with exactly the same names, the new definitions will overwrite the old ones and the document will be entirely reformatted.

It is unlikely that the new document layout will be perfect in every detail, but much tedious recomposition work has been eliminated and you are left with only the details to check.

### Key FrameMaker features which aid Acrobat document composition

- **MULTIPLE MASTER PAGES:** Whereas Ventura allows only a left and a right master page, you may have more than twenty in a FrameMaker document. Different background colours, text and art, applied efficiently by applying named master page layouts to any desired page, can help provide navigational clues to users of a screen document.
- **VARIABLES IN PAGE HEADERS:** Variables are a very powerful feature of FrameMaker. For instance, you can set up variables in a page header to pick up automatically and concatenate the names of the current section, chapter and subheading, as recommended on page 8.
- **GENERIC COLOUR DEFINITIONS:** If you set up spot colours under generic names such as ‘Background panel’ or ‘Red’, you can transform the definition of these between alternate formatting templates.
- **GENERIC CHARACTER FORMATS:** Whereas Ventura Publisher uses typographic stylesheets only at the level of the paragraph, FrameMaker has stylesheets for exceptional character formats. For instance, this document uses defined character formats for expert-set numerals (3½), ACRONYMS, **emphasis**, *italics* and `computer text`.

  Defining character formats generically lets you do things like globally converting all emphasized text to red for a screen-based document.
- **SPECIAL HEADINGS FORMATS:** a heading may optionally be defined as a **sideheading** (flanking the text flow) or **run-in heading** (with no linebreak, so text continues on the same line). Conversion between ordinary headings and sideheadings is valuable for documents being reformatted between landscape and portrait display.
- **CONDITIONAL TEXT:** Sometimes text is required in one version of a document, but it should be suppressed in another version. For instance, you may wish explicit ‘see page x’ cross-references suppressed in an on-screen document, in favour of hyperlinks activated by clicking on a word.

  In FrameMaker, text may be given a **condition tag**. By deciding which conditional text shows in which document, you can keep all of the text in the same authoring document but express/suppress it for specific formatting instances.
- **CONDITIONAL GRAPHICS:** If pictures are imported by reference into frames anchored to the text, those frames may also be given condition tags. By using alternate condition-tagged frames anchored to the same location, you could have high-resolution greyscale versions of pictures in a paper-oriented document, but substitute lower-resolution colour versions for an on-screen document.

- **GENERIC TABLE FORMATS:** FrameMaker is particularly good at setting tabular material, *e.g.* timetables. By setting up named table formats, defined differently in alternate templates, you can use colour fills in a screen document and tints in a printed document, thinner rules in a paper document and more definite ones in a screen document, for example. (The ruling styles used for tables in a document are generically defined by name as well.)
- **AUTO-UPDATING OF CROSS-REFERENCES:** Cross-references can be built within and between FrameMaker documents. The cross-reference is given a format (e.g. '*see table* `<table number>` *on page* `<page number>`') and is aimed at a particular target —in this case, a table.

  Documents converted using alternate templates are inevitably repaginated, but in FrameMaker these cross-references can be automatically updated, reflecting not only the new pagination but also, if necessary, new content for the target (for instance if a subtitle referred to in a cross-reference has been edited.)

  FrameMaker version 4.0.2 turns cross-references into Frame hypertext links and also exports these links to Acrobat through the pdfmark function.

## Some productivity aids for authors and publishers

Additionally, FrameMaker provides facilities which assist efficient editing and revision control:

- **TIME/DATE STAMPING:** 'System variables' can be used to stamp the date and time of production of a particular version of a document during the production cycle, and the filename can also be recorded. You might locate these in a footer, for instance. They could by defined as conditional text so that they could be suppressed for production versions of the document.
- **COMMENTS:** Conditional text can be used to send comments between collaborating authors. Again, these can be stripped out for production versions.
- **CHANGE BARS:** These may be set to show which parts of the document have changed during the current editing cycle. They can be cleared at the end of one revision cycle and restarted for the next. (Being printable elements, change bars can be included in Frame-generated Acrobat files too.)
- **FRAMEMAKER HYPERTEXT:** FrameMaker has its own facilities for producing hypertext documents, rather more powerful than Acrobat's.

  I do not propose to make comparisons here, but merely comment that FrameMaker hypertext links (such as those automatically generated between cross-references or from entries in a Table of Contents or Index) are useful during the editorial process for checking the integrity of such links.
- **FULL-FEATURED MARK-UP LANGUAGES:** The **Maker Mark-up Language (MML)** and the **Maker Interchange Format (MIF)** are 7-bit ASCII document description languages which allow database report generators and other similar programs to produce generically marked-up text which can be flowed into a FrameMaker template, removing the need for manual tagging. Using MIF a complete Frame document can be generated without FrameMaker!

  Custom publishing environments could be built in which database-originated text (*e.g.* directories) or generically-tagged texts in SGML are converted in batch mode to FrameMaker MIF files, which on conversion to a formatted FrameMaker document then require only a quick check before being sent out for Acrobat distillation.

## Discipline and imagination are essential!

From all of the foregoing it can be seen that to get the best out of Acrobat, as with any publishing technology, requires the input of graphic designers who are aware of the reader's situation and are prepared to get their hands dirty working out efficient production strategies.

I am aware that this paper has raised more questions than it has given answers—but that is not so very inappropriate when we stand at the start of a new era in publishing, where new criteria and production methods are yet to be fully determined.

There is much to learn.

**Conrad Taylor may be contacted via email as conrad@ideograf.demon.co.uk**