Tables and Graphics II: What Works Effectively in Digital Formats?

Speakers:
G Paul Bozuwa
Carol Hollenbeck
Capital City Press
Montpelier, Vermont

Irina Golfman
Irnera Associates*
Newton, Massachusetts

Reporter:
Lee N Miller
Ecological Society of America
Ithaca, New York

As publishing progresses toward all-electronic inputs of text and graphics, the compositor must somehow turn these inputs into a form that will produce print and/or electronic outputs. The entire publishing process can stumble, stutter, or stop at this input-output transition point if inputs provided by the author and modified by the editor can't be handled efficiently by the compositor.

In "Tables and Graphs II," Paul Bozuwa, Carol Hollenbeck, and Irina Golfman explained what works and what doesn't work when editors send graphs and tables to the compositor in electronic form for use in print and digital publications.

The graphics portion of this presentation focused on line art used in graphs. Like text, graphics may be supplied by authors in printed form or in dozens of electronic formats created by graphics applications or by scanners. All of these must be converted ("normalized") to a single standard format required by the printer's page-description application or by World Wide Web protocols. Bozuwa compared the quality of images resulting from different conversion processes.

When graphs are printed out by an author at 300 dpi and subsequently scanned on a high-end flatbed scanner, the resulting TIFF (Tagged Image File Format) file images have somewhat jagged lines. In contrast, graphics programs that produce EPS (Encapsulated PostScript) files have sharper images.

File size also differs between formats, and this affects the amount of space taken up in an archive and the amount of time required to transmit a file. The scanner-output file is large because every bit in the image must be specified as a 1 or a 0 (black or white). EPS files, however, describe data as vectors, in much the same way that scalable fonts are handled. This allows file sizes to be smaller by a factor of about 15.

A further advantage can be gained by converting fonts (used for lettering of axis labels, scales, and keys) to paths. This increases file sizes somewhat, but avoids the necessity of providing font files along with the graphic files. Fonts were characterized as "the bane of the digital environment".

The use of scanning to create electronic files requires that authors provide sharp prints on paper of graphs that were probably produced in electronic form originally. How ironic! Authors may also provide digital versions of the graphs, but these will be ignored if their formats are incompatible with the "normalized" system used by the compositor and printer.

If the final publication will be on the World Wide Web, graphics must be in GIF (Graphics Interchange Format) rather than TIFF or EPS formats. Thus, another conversion (from EPS to GIF) will be required, by use of an application such as DeBabelizer or Image Alchemy.

The key to efficiency and image quality is to establish a workflow that standardizes the formats of all graphs. If authors are instructed about the compositor's requirements and editors work with them to achieve compliance, a smooth operation of the system can result.

Tables present a more formidable challenge than graphs do in the electronic environment. The first challenge is in the ways that authors record tabular data in word processors, spreadsheets, or database managers. If they use pull-down menus or combinations of keystrokes to access the 128 special ASCII characters not found on standard keyboards, these characters may not be recognized by the compositor's software. Some word processors treat special characters as graphics, and these can't be reproduced by typesetting systems.

Another problem in table construction results from the invisible embedded software commands created by the author's word processor, including those inserted when a table is typed with "columnar mode" turned on. Tabular data destined for electronic processing should always be typed in horizontal rows rather than in columns. Other problems are caused when authors separate columns with spaces rather than tabs, when they insert vertical column rules or straddle heads, or have runovers within columns. All of these complications will require adjustments by the compositor, who needs a hard-copy printout of the table for reference.

If a work is to be published electronically, either exclusively so or in addition to a printed version, tables will need to be prepared with SGML (Standard Generalized Markup Language) or HTML (HyperText Markup Language) coding. In each case, the markup must conform to a DTD (Document Type Definition). The DTD describes the hierarchy of the parts of a document. For example, a DTD for a technical article will have an article as its top-level element. The article element might include 3 components: one for front matter, one for the body of the article, and one for back matter. Each of these 3 elements will have components of their own. For example, a back-matter section might include acknowledgments and bibliography. In this sense one can describe a DTD as representing the document in a layered manner. Rather than creating a new DTD, it is generally preferable to use a publicly available one that nearly suits your purposes, modifying it as necessary to fit your publication.

A whole new set of problems is encountered when dealing with tables on the com-

formerly JIB Associates

puter screen. Many tables are simply too large to view on a single screen, and it may be difficult to recall the sequence of column and row heads when reading data deep within the table. Among the solutions to this problem are the following: 1) Offer the table as a graphic, to be printed out rather than read on the screen. 2) Offer the table as a graphic displayed at a small size on screen, but provide tools for zooming in to read enlarged portions of the table, as well as tools for scrolling. 3) Allow the data to scroll independently of row and column heads, so that the heads remain visible while viewing different parts of the data field. (Tools for doing this have not yet been developed.)

Graphic models avoid the need to convert a table to SGML format, but without SGML one cannot search within the table or manipulate the data. A geometric model is the more common means of presenting data electronically. Each datum is placed in a cell with its row and column location described by its place in the sequence. This approach, however, requires that all tables follow the format described in the DTD, which may prove inappropriate in some cases.

A more flexible approach is a "content-based table model" that describes the relationship between data sets, but not the characteristics of presentation. It is very difficult to convert automatically from a typeset to an SGML representation of such a table, and the result may not replicate the original presentation.

Tables described in HTML can be viewed by using standard Web browsers. If the table is SGML coded, a special SGML viewer is needed. One example of an SGML viewer is Panorama Pro by SoftQuad. Unfortunately, this viewer can't deal with every DTD, so it may require tweaking to get it to work properly. A more sophisticated (and much more expensive) viewer is EBT (Electronic Book Technology). The simplest solution may be to convert the SGML document to the simpler HTML, and then view it in a Web browser.

In her summary, Golfman noted that "many electronic presentation issues for tabular material are yet to be solved."

This well-organized session was particularly appropriate for the newcomer to issues of electronic handling of tables and graphs. It was accompanied by a very complete and clear set of handouts that paralleled the presentation. The talks offered a compositor-centric view of the subject. Since compositors sit at the pivotal transition point between input and output, authors, editors, and publishers must learn how to provide inputs that these folks can handle efficiently.

ANNOUNCEMENT

BELS Examination Schedule

During 1997, the certification examinations administered by the Board of Life Science Editors (BELS) will be given from 1-4 PM according to the following schedule:

15 March, Saturday, New Orleans, Louisiana
19 April, Saturday, Burbank, California
3 May, Saturday, Philadelphia, Pennsylvania (CBE annual meeting)
24 May, Saturday, Helsinki, Finland (EASE conference)
12 July, Saturday, Atlanta, Georgia
11 November, Tuesday, Boston, Massachusetts (AMWA conference)

For information and an application form, write to BELS, PO Box 824, Highlands NC 28741-0824.