Prototyping Telephone-directory Pages with TEX

Richard Southall

Leckhampstead, Buckinghamshire MK18 5NZ, UK

Abstract. *The development of a prototype formatter for telephone-directory pages, written in TEX and using fonts made with Metafont, is described. The formatter was used to decide the detailed typography of directory entries. Issues connected with the markup language used in the directory data files are discussed.*

1. Introduction

In 1995 I was approached by Ladislas Mandel, a well-known French designer who has produced many typefaces for telephone directories, to work with him on the production of a family of new designs for US West Direct[1] (USWD), a directory publisher in the western United States. The considerations that underlay my decision to specify the shapes of the new characters as Metafont programs, rather than using the more usual technique of contour capture followed by PostScript or TrueType font generation, are discussed elsewhere.[2]

In the early part of the project we used the TEX-based tools that I had developed at Stanford and Strasbourg in the 1980s for testing the individual members of the new family as they were developed. In mid-1996, when the stage had been reached at which working Metafont programs were available for all the designs, we began to feel the need to test the family as a whole in realistic contexts. This was particularly necessary because of the variety of different formats, many more than in normal European practice, in which USWD's directories were published. USWD sent me 5000-entry extracts from typical data files, and I began to look at the problems of formatting them.

---

1 Now US West Dex.

2 Richard Southall, "Metafont in the Rockies: the Colorado Typemaking Project", *Electronic Documents, Artistic Imaging and Digital Typography*, (R.D. Hersch, J. André, and H. Brown eds.), EP98 and RIDT98 proceedings, Springer-Verlag LNCS 1357, Heidelberg 1998, 167–180.

2. Data types and structures

The project, which along with the new typeface family had acquired the name Colorado in November 1995, was concerned only with white-pages directories. USWD is obliged to supply these to the users of its sister company's telephone services, and they are not a source of profit for USWD even though some kinds of entry can be charged for. Since by far the largest elements of their production costs are paper and printing, the objective of any redesign must be to combine the maximum ease of use with the minimum page count for a given body of data.

There are three different types of white-pages listing: residential, business and combined (in which residential and business entries are run together in the same listing). Generally speaking, combined listings are more often used in directories for smaller communities. All the directories have a section at the beginning which contains listings for government organizations at the federal, state, county and city or township level. Thus there are four different types of white-pages data: residential, business, combined and government. Although all the data files have the same basic structure of a front-matter section followed by a list of entries, the detail of the structures that occur within entries differs substantially from one type of data to another. I began with residential listings, both because these seemed to be the most straightforward and because they were the ones for which Mandel had worked out the visual relationships between variants in the new family of designs.

Although there are many varieties of entry structure in the different data types, they are all derivatives of a basic pattern. An entry type designator is followed by a sequence of data objects, terminated by an end-of-entry marker. In residential entries the objects are mostly delimited and numbered fields, and the data for an entry looks like this:

```
<amrtrl><fe1>Doe<mc><fe2>Jane<mc><fe5>455 N Main<mc>
<fe6>Middletown 55667<mc><fewl><fe8>456-7890<mc><feem>
```

Here `<amrtrl>` designates a simple residential entry. Fields 1 and 2 are surname and given name; fields 3 and 4, title and qualifications, are omitted in this instance; field 5 is the street address, field 6 the locality name and postal (zip) code, and field 8 the telephone number. Field 7, once again omitted in this instance, would hold additional dialling information if the number in field 8 were outside the local calling area. The `<fewl>` object denotes leader characters.

The next entry in the listing might look like this:

```
<amrtrlsss><fe1>Doe<mc><fe2>John<mc><fe5>692 S Broadway<mc>
<fe6>Middletown 55668<mc><fewl><fe8>456-9876<mc><feem>
```

Here, because Jane and John Doe share the same surname, its repetition is suppressed by the `<amrtrlsss>` entry type designator, and the entry in the printed listing begins with John's given name.

Mandel had conceived the new typeface family in terms of a strict allocation of design variants to the different semantic components of residential entries. Surnames and telephone numbers were to be set in semibold, given names in medium, and all the elements of the address in light condensed. Within this rigid framework, though, there was still scope for variation. His very first requirement for the prototype formatter was that it should allow him to test the effect of putting the zip code ahead of the locality name, since he was concerned about the consequences for usability of having two groups of numerals – the zip code and the telephone number – next to one another in the printed entry. If locality and zip had been in separate fields in the data, this would not have been difficult. As it was, I had to figure out a way of splitting data field 6 into its components and inverting their sequence on request.

Solving this problem, and the associated one of interpreting the data files, gave me most of the components I needed to build a general-purpose parametrized entry-formatting engine. The listings contained occasional character sequences that were very unTEXlike–`American Legion Post #600` and `Cellular %` (for a cellular-telephone supplier) are both real examples – and I felt it was important not to modify the data files. I juggled `\catcode`s and made a scanner that looked at each character in the input stream. Similarly, when the contents of the name field overflowed the first line of a printed entry I needed to know how much space was left on the second line, so I made a word-by-word line assembler that in its final form used a variation on the list macros in Appendix D of *The TEXbook*. It was not hard to build switches into the formatter which, according to their settings, would omit the zip code altogether; capitalize the locality name; put leader characters between street address and locality, between locality/zip and telephone number, or not at all; set indents or dittos as markers for repeated surnames; and so on.

The beginning of a typical driver file came to look like this (explanatory comments follow `%%`):

```
% fargo_mr.tex
% Driver file for Fargo-Moorhead residential pages
                       %% North Dakota - Minnesota borders
\input liststyl        %% formatting macros
\remark Fargo-Moorhead residential, new style, max page
                       %% remark appears in page footline
% Formatting switches  %% all off for this example
                       %% comments give their effects when on
%\noloc                %% locality name suppressed
```

```
DOE Jane 455 N Main Middletown 55667 ............ 456-7890
   John 692 S Broadway Middletown 55668 .......... 456-9876

DOE Jane 455 N Main ............ Middletown 55667 456-7890
   John 692 S Broadway .......... Middletown 55668 456-9876

DOE Jane 455 N Main ............ 55667 MIDDLETOWN 456-7890
 » John 692 S Broadway .......... 55668 MIDDLETOWN 456-9876

DOE Jane 455 N Main ............ MIDDLETOWN 55667 456-7890
 » John 692 S Broadway .......... MIDDLETOWN 55668 456-9876
```

Figure 1 – Variations in residential entry formatting.

```
%\localcaps             %%               capitalized
%\addrwithloc           %% leaders between locality/zip and number
%\zipfirst              %% zip code before locality name
%\nozip                 %%          suppressed
%\dittoson              %% dittos mark repeated surnames
%\leaderson             %% leader characters set
%\headlinecaps          %% page headline in all capitals
```

Setting the appropriate switches gave us variations like those in Figure 1, extended over several pages of real directory content so that we could get a good idea of their effects. The final variation shown, with locality and zip in their natural order,[3] the locality capitalized and dittos marking repeated surnames, is the one with which almost all the subsequent development work was done.

3. Microtypography in directory entries

Mandel's choice of weights and widths for the different variants of the Colorado family was intended to facilitate a particular model of the task of searching for a directory entry. The assumption is that users know the surname and given names in the entry they are looking for. The first landmark in a search is the two surname/initial pairs in the directory page headline. Having found the correct page, the user scans the bold capitalized surnames at the left edge of successive columns. If the surname is unique in the directory, the search is finished when it is located. If it is not, ditto marks show the range over which it is repeated, and the user scans given names within the range. Given names are set in a lighter weight than surnames, to make a visual distinction between the two. The street address, considered to be secondary to given names as a search criterion, is set in light condensed type. This is also used for locality

[3] Natural, that is, in the culture for which the directories were intended.

name and zip code; in the new design, the capitalized locality name provides an alternative to given names as a search criterion. The telephone number is set in the same bold variant as the given names, to provide a clear visual delimiter at the right edge of the column.

Given this broad picture, the fine detail of graphic structure in the printed entry still had to be developed. My objective in doing this was to reflect the semantic structure of the entry as closely as possible in its visual structure, while keeping the horizontal extent of the entry at a minimum.

As suggested above, a simple residential entry is made up of four semantic objects, closely related to the delimited fields in the data:[4]

$\langle$entry$\rangle$ $\rightarrow$ $\langle$name group$\rangle\langle$street address$\rangle\langle$locality group$\rangle\langle$telephone number$\rangle$

These objects need to be clearly distinguished in the visual structure. The choice of typeface variant provides first-level differentiation in the simple case, but not every case is simple. For example, the $\langle$name group$\rangle$ in the definition above expands further:

$\langle$name group$\rangle$ $\rightarrow$ $\langle$surname$\rangle\langle$given names$\rangle$[$\langle$title$\rangle$[$\langle$qualifications$\rangle$]]

Titles are character groups such as `Dr`, `Rev` or `Col`; while qualifications are most often short groups such as `MD` or `DDS`, they can occasionally occur as long phrases such as `Endodontics Associates Limited` (also a real example). Both kinds of object are set in the same typeface variant as the street address which immediately follows them, and there needs to be some visual (and hence graphic) separation to reflect the semantic distinction between the two.

Because the contents of the different data fields were handled separately in the formatter, it was easy to parametrize the horizontal spaces between them. Also, because we were making our own fonts with Metafont, we could choose our own values for interword spaces. For example, I felt that the graphic objects that occurred in street addresses or title/qualification strings were already well demarcated visually by their content – groups of numerals, capitalized words, and capital letters either alone or in small groups. Hence the interword space for the light condensed variant could be much narrower than normal: 0.20 em in the event, as opposed to the 0.24 em of the medium or Computer Modern's 0.33 em. This allowed us to satisfy a general principle, that the spaces between the graphic renderings of semantic objects should always appear wider than the spaces within them, while still keeping control of the entry's horizontal extent.

4 In this notation the names of semantic objects are set in roman type inside angle brackets $\langle\rangle$. Objects enclosed in brackets [ ] are optionally present; objects in braces { } are optionally repeated.

4. Column width and vertical extent

It hardly needs to be said that for a given amount of data the way to reduce the number of pages in a directory, and hence its paper and printing costs, is to maximize the number of entries on each page. The straightforward way to do this is to increase the number of lines in each column by reducing the nominal size of the type, and hence the vertical distance between the baselines of successive lines in the printed listing. The limit to this approach is set by the type itself; for any design, there is a size at which the character images become undecipherable. Mandel's speciality for many years has been to make typefaces that are exceptionally legible in small sizes, such as the 3.3 Didot (3.5 pt)[5] design used in the French A5-format *mini-annuaires*. The Colorado family was intended to be used at a nominal size of 5 pt, with a baseline separation of 5.5 pt.

This was already less than the 6.25 pt used by USWD in the 7×9-inch 3-column residential listing with whose data we began developing the new typography, so we were disagreeably surprised when our first tests yielded vertical extents that were *greater* than the existing ones (although the appearance of the page was enormously improved). It turned out that, contrary to most European practice, entries containing two given names spelt out in full were very common in the listing. Since the medium-weight variant of Colorado used for given names in the new design was much wider in set than the narrow bold used in the existing design for both given names and surnames, almost every such entry divided after the name group to make two lines.

The solution, which seemed strongly counter-intuitive at the time, was to increase the column width and thus reduce the number of divided entries. Changing from three columns of 144 pt (12 picas) to two columns of 220 pt (18 picas 4 pt) decreased the average number of lines per page by a third from 307 to 205, but *increased* the number of entries by nearly 3.6 per cent, from 195 to 202.[6] This represented a saving of around 21 pages in the 600 pages of the existing residential listing, or nearly 2.5 million pages over the whole print run of the directory.[7]

The same approach, formalized as Callison's Principle after the project director at USWD who first articulated it,[8] helped to compensate for the inexorable increase of entry content in metropolitan-area directories. These were already

5 Point dimensions are in Anglo-American (pica) points.

6 All line and entry counts are averaged over 10 pages.

7 This figure assumes that all of the 117 000 residents in the listing received at least one directory each.

8 Callison's Principle states that 'The wider the column, the more you get in.'

very large books: the existing residential directory for the Denver metropolitan area has more than a thousand 9 × 11-inch pages, with around 470 entries per page in four columns. The number of telephone connections increases at about 10 per cent each year, and because of this it becomes necessary for the telephone-services suppliers to define new area codes. In many metropolitan areas the new codes are 'overlays', so that a geographical region that used to be served by a single code now finds itself saddled with two or more. This in turn obliges the directory publishers to add an area code to every entry in all the listings for the region. There is also strong pressure from marketing departments to include zip codes in listings, since postal addresses are incomplete without them and the telephone directory is a reference source for addresses.

Set with the Colorado family in 5 pt nominal size with 5.5 pt baseline separation, Denver residential data in four columns without zip codes or area codes averages 512 lines and 465 entries per page. Adding zip codes in the four-column format reduces the entry count to 384; adding area codes as well reduces it further to 323, still with 512 lines per page. Changing from four to three columns reduces the line count per page to 383; the entry count, with zip codes and area codes included, increases from 323 to 372. Once again, the format with fewer columns is more economical in total extent than the one with more.

5. Markup in the data files

Although I have been unable to discover the history of the markup notation used in USWD's data files, I get the strong impression that it evolved in much the same way as our formatter did: beginning with simple cases, and extending itself with more or less difficulty to handle more complicated ones as they were encountered. Residential entries, without and with suppressed surnames, are the simplest objects in the data. The next simplest are residential entries with an additional line:[9]

```
<amrtrl><fe1>Atchison<mc><fe2>Robert H<mc><fe5>2238 229 Av
NW<mc><fe6>Oak Grv 55005<mc><fewl><fe8>753-1857<mc><feem>
<amrtrlsss><fe1>Atchison<mc><fe2>Steven & Alison<mc><fe5>
16749 NE Washington St<mc><fe6>Ham Lk 55304<mc><fewl>
<fe8>434-4468<mc><feem>
<amrtrlal><eqin1><fe1>Teenagers<mc><fewl><fe8>434-2162<mc><feem>
<...>
```

9 All the remaining examples are taken from published data for the northwestern suburbs of Minneapolis-St Paul.

ATCHISON Robert H 2238 229 Av NW .... OAK GRV 55005 **753-1785**
» Steven & Alison 16749 NE Washington St .... HAM LK 55304 **434-4868**
Teenagers **434-2126**
ATCHLEY Pat 13230 Meadowood Trl .... COON RPDS 55448 **427-3443**

BALDWIN Jack R 770 129 Av NE .... BLAINE 55434 **767-4575**
Teenline **767-9692**
» James D 416 2 Av NE .... OSSEO 55369 **424-3234**

Figure 2 – Additional-line entries in a residential listing.

ANDERSON Kurt D MD—
Orthopaedic Surgery
Telephone Answered 24 Hours
290 Main St NW .... ELK RVR 55330 **441-9298**
» Kurt & Kris 14369 202 Av NW .... ELK RVR 55330 **441-2585**

Figure 3 – Extra lines in an additional-line entry.

```
<amrtrlsss><fe1>Baldwin<mc><fe2>Jack R<mc><fe5>770 129 Av NE<mc>
<fe6>Blaine 55434<mc><fewl><fe8>767-0475<mc><feem>
<amrtrlal><eqin1><fe1>Teenline<mc><fewl><fe8>767-9622<mc><feem>
<...>
```

Here the `<amrtrlal>` entry type designator introduces the additional line. In the existing design the `<eqin1>` object generated an extra indent, but in the original version of the new design (Figure 2) this was omitted. Additional-line designators suppress capitalization of the contents of `<fe1>` in residential listings. The visual distinction between the medium weight of given names and the semibold of `Teenagers`, or between capitalized full-out surname and uncapitalized indented `Teenline`, seemed to me to differentiate adequately between the additional line and the components of the main entry.

As the semantic structure of entries becomes more complex (Figure 3), so does the data:

```
<amptrl><fe1>Anderson<mc><fe2>Kurt D<mc><fe4>MD<piem><mc><feem>
<amptrlal><feel><eqin1>Orthopaedic Surgery<mc><feel><eqin1>
Telephone Answered 24 Hours<mc><eqin1><fe5>290 Main St NW<mc>
<fe6>Elk Rvr 55330<mc><fewl><fe8>441-0298<mc><feem>
<...>
```

(The `<amp...>` prefixes introduce professional entries in a residential listing; the `<piem>` object in field 4 at the end of the `<amptrl>` entry is an em rule.) Here, as in the residential examples, the additional-line entry is delimited in the data by `<...al>` and `<feem>`; but there are also extra lines in the printed entry, corresponding to the objects delimited by `<feel>` and `<mc>` inside the entry data. Although this markup is procedural rather than semantic, its notation is

ABEL CHIROPRACTIC ASSOCIATES
646 East River Rd .... ANOKA 55303 **427-7689**
Abel Electrical Contractors Inc 17701 149 Av N .... DAYTON 55327 **323-1106**

Figure 4 – Varying graphic treatments in business listings.

a mnemonic for the real-world distinction between *additional lines*, which are hardware features of the telephone company's world, and *extra lines*, which are graphic features of a directory entry.

Not surprisingly, the markup language seems to be at its most expressive when it is dealing with things that are important to the telephone company or the directory publisher. By far the largest element of its vocabulary are what I have called entry type designators. In the publisher's terminology these are UDACs – universal directory advertising codes – and there are around 60 different ones in USWD's white-pages data. Roughly three-quarters of these occur in business listings, where many types of entry are chargeable.

The UDAC that introduces an entry determines the graphic treatment of its major components. For example, in Figure 4 both entries have semantic structures very like that of a normal residential entry, but the first one's UDAC—w*hite-pages* b*old* l*isting and* n*umber*—means that the contents of the name field are capitalized and set in large bold type along with the telephone number:

```
<amwbln><fe1>Abel Chiropractic Associates<mc><fe5>646 East River
Rd<mc><fe6>Anoka 55303<mc><fewl><fe8>427-7689<mc><feem>
<ambtrl><fe1>Abel Electrical Contractors Inc<mc><fe5>17701 149 Av
N<mc><fe6>Dayton 55327<mc><fewl><fe8>323-1106<mc><feem>
```

Below the level of the UDAC, the only elements of the language with semantic significance are the <fe*>...<mc> field delimiters. These are mostly, but not always, used consistently in the data: <fe1> begins a name, <fe6> a locality designator, <fe8> a telephone number, and so on. The other elements of the language afford procedural rather than semantic markup. The <feel>...<mc> construction generates extra lines in the printed entry, but these, as Figure 3 demonstrated, do not necessarily realize objects of the same semantic type. The <eqin*> mechanism is merely a way of producing hierarchies of indents, which are not always effective as a means of expressing semantic hierarchies.

6. Government pages

Some of the entries in the government pages at the beginning of the directories present difficult problems for any formatter. Figure 5 shows abbreviated excerpts from the listings for two departments in a county organization.

ANOKA COUNTY OFFICES—
2100 3 Av ANOKA 55303
GENERAL INFORMATION **421-4670**

LIBRARIES—
Administrative Offices 707 Hwy 10 NE .... BLAINE 55434 **780-1463**
Anoka Public Library 2135 3 Av N .... ANOKA 55303 **421-5080**
Centennial Branch 100 Civic Heights Cir .... CIRC PINES 55014 **786-5210**
Round Lake Branch Andover Center
13855 Round Lake Blvd NW .... ANDVR 55304 **427-7305**
Rum River Branch Rum River Plaza 5910 167 Av NW ... RMSY 55303 **753-3396**
St Francis Branch 3519 Bridge St Nw .... ST FRANCIS 55070 **753-3121**

LICENSE BUREAU—
Hours M-F 8 AM-5:30 PM Sa 8 AM-2 PM
(Coon Rapids Open T & Th Until 6:45 PM)
2100 3 Av .... ANOKA 55303 **421-4670**
(St Francis Closed M-F Noon-1 PM)
General Information **323-7577**
Anoka License Center—
Old Milk Factory
2665 4 Av N Anoka 55303
Blaine License Center—
11000 Hwy6 85 NE Blaine 55434
St Francis License Center—
St Francis Library Building—
3519 Bridge St NW St Francis 55070
Auto Licenses
Bicycle License
Hunting & Fishing Licenses (Excludes St Francis)
Motor Vehicle Registration
Watercraft Registrations
MEDICAL EXAMINER **422-6413**

Figure 5 – Entry structures in government pages.

The semantic structures of the listings for the libraries and the license bureau are very similar:

$\langle$department listing$\rangle \rightarrow [\langle$opening-hours information$\rangle]\langle$branch list$\rangle$
$\langle$branch list$\rangle \rightarrow \langle$branch information$\rangle\{\langle$branch information$\rangle\}$
$\langle$branch information$\rangle \rightarrow$
$\quad\langle$branch name$\rangle[\langle$branch location$\rangle]\langle$address$\rangle[\langle$telephone number$\rangle]$

but their graphic structures are quite different, because the entries are so differently marked up:

```
<amgtrlal><eqin2><fe1>Rum River Branch Rum River Plaza<mc><fe5>
5910 167 Av NW<mc><fe6>Rmsy 55303<mc><fewl><fe8>753-3396<mc><feem>
<amgtrlal><eqin2><fe1>St Francis Branch<mc><fe5>3519 Bridge St
Nw<mc><fe6>St Francis 55070<mc><fewl><fe8>753-3121<mc><feem>

<amgtrlal><feel><eqin3>Anoka License Center<piem><mc>
<feel><eqin4>Old Milk Factory<mc>
<feel><eqin5>2665 4 Av N Anoka 55303<mc>
<feel><eqin3>Blaine License Center<piem><mc>
<feel><eqin4>11000 Hwy6 85 NE Blaine 55434<mc>
<feel><eqin3>St Francis License Center<piem><mc>
<feel><eqin4>St Francis Library Building<piem><mc>
```

ANOKA COUNTY OFFICES—
2100 3 Av ANOKA 55303
GENERAL INFORMATION **421-4670**

LICENSE BUREAU—
Hours M-F 8 AM-5:30 PM Sa 8 AM-2 PM
(Coon Rapids Open T & Th Until 6:45 PM)
2100 3 Av .... ANOKA 55303 **421-4670**
(St Francis Closed M-F Noon-1 PM)
General Information **323-7577**
Anoka License Center—
Old Milk Factory 2665 4 Av N ANOKA 55303
Blaine License Center—
11000 Hwy 65 NE BLAINE 55434
St Francis License Center—
St Francis Library Building 3519 Bridge St NW ST FRANCIS 55070
Auto Licenses
Bicycle License
Hunting & Fishing Licenses (Excludes St Francis)
Motor Vehicle Registration
Watercraft Registrations
MEDICAL EXAMINER **422-6413**

Figure 6 – License-bureau data with modified markup.

```
<feel><eqin5>3519 Bridge St NW St Francis 55070<mc>
<feel><eqin6>Auto Licenses<mc>
<...>
<feel><eqin6>Watercraft Registrations<mc>
<eqin1><fe1>Medical Examiner<mc><fewl><fe8>422-6413<mc><feem>
```

(I have added linebreaks to the license-bureau data to help elucidate its structure, and omitted some repeated structures to save space.)

Each branch library has its own additional-line entry. The whole list of license-bureau centres, on the other hand, beginning with `Anoka License Center`, is a single entry which ends, not with `Watercraft Registrations`, but with the medical examiner's telephone number! All the graphic structure within the entry is produced by the procedural `<feel><eqin*>` mechanism rather than by semantic markup.

The license-bureau data could be modified to have the same semantically-oriented markup as the libraries data (Figure 6), but of course this is not possible in practice. As with any database-publishing application, formatter and typographer are at the mercy of the data and all its inconsistencies, for whatever reason these might exist.

7. Conclusions

Building the formatter was a non-trivial task which has been more than justified by the usefulness of the product. At the beginning, Mandel and I had envisaged it as nothing more than a tool for modifying the graphic structures of simple data. It turned out, however, that the real data was not simple. By the time I had succeeded in interpreting and formatting the first residential listing, tackling the business and combined data involved little more than the construction of formats for successive UDACs. Interpreting government data was no more difficult; formatting it sensibly was (and is) another matter, as Section 6 suggests.

The formatter's greatest value has been as a communication tool, both between members of the project team in Denver and Omaha and myself in Leckhampstead and between the team as a whole and the rest of USWD. With the same data sets present on USWD's real machines and mine,[10] the identity of the TEX/Metafont virtual machines meant that we could generate alternative fonts and formats, and discuss them by e-mail or telephone, with no uncertainties about what we were discussing; and if modifications needed to be made, only small files had to be exchanged.[11] Equally, if it became necessary to convince other departments in USWD of the correctness of our ideas, we could change parameters or reset switches in a driver file and come back into the meeting within a few minutes, with half-a-dozen pages of reformatted material.[12]

An unanticipated, but in the event extremely valuable, use of the formatter was to generate statistics such as those presented in Section 4. It would have been impossibly tedious to arrive at the figures manually. A very simple modification to the TEX code adds up lines and entries (defined as data objects ending with a telephone number), maintains running averages, and writes the results to the log file. In the same way, the formatter will now report the values of all its typographic parameters, to help in the task of implementing the new design with USWD's own page-makeup software.

In conclusion, I would strongly recommend a similar approach to anyone confronted with the problem of deciding on the detailed typographic design of extensive complex lists. The power and flexibility that a fully-parametrized formatter affords to the typographer is well worth the effort involved in building it.

Acknowledgements

As always, my thanks go to Bob Callison of USWD and the members of the Colorado project team.

[10] Security considerations, as well as connect-time costs, prevented me from working directly on USWD's machines.

[11] Driver files were around 2 kbyte; the whole formatter was less than 68 kbyte. By contrast, the smallest of the PostScript files for the illustrations in the present paper is 23 kbyte.

[12] This did not always work out exactly as planned. The use of dittos to mark the range of repeated surnames was unilaterally vetoed late in the project by the company's chairman.