

A Macro Package for View Graphs and Slides

T. A. Dolotta
D. W. SmithBell Laboratories
Murray Hill, New Jersey 07974

1. INTRODUCTION

This manual describes a package of UNIX† *troff*(1)¹ macros called MV, designed for typesetting view graphs and slides. This manual assumes that the reader has a basic knowledge of the UNIX system, the UNIX editor *ed*(1), and *troff* [3,8].

With these macros, one can easily prepare view graphs in a variety of dimensions (see Table I below), as well as 35mm slides and 2×2 “super-slides.” These transparencies can be made in a variety of styles, in different fonts, with oversize titles, and with highlighted subordination levels. Because the text from which the foils are typeset is stored on UNIX, the contents of a foil can be readily changed to include new data, or can be incorporated into a new presentation; the text of the foils can be passed through *spell*(1), preprocessed by *eqn*(1), *tbl*(1), *cw*(1), etc.

It is not possible to include artwork, graphics, or multicolored text in foils made with this package except by manual cut-and-paste.

2. SIMPLE EXAMPLES

Before explaining the macros in detail, we illustrate the formatting process with two examples.

2.1 Trivial Example

If you are familiar enough with the UNIX editor *ed*(1) to create the following text file, naming it *trivial*:

```
.Sw
Six stages of a project:
.B
wild enthusiasm
.B
disillusionment
.B
total confusion
.B
search for the guilty
.B
punishment of the innocent
.B
promotion of the non-participants
```

and if you then utter the following UNIX command:

```
mvt trivial
```

you will be rewarded with the first view graph in the Appendix. The *.Sw* is a *foil-start macro*; by looking at that view graph, you should be able to figure out what the *.B* macro does.

† UNIX is a trademark of Bell Laboratories.

1. The notation *name*(*N*) indicates entry *name* in Section *N* of the *UNIX User's Manual* [2].

2.2 Less Trivial Example

The foil that results from typesetting the following input is the second view graph in the Appendix.²

```
.Vw 2 "Less Trivial" "June 29, 1980"
.T "What the Walrus Said"
``The time has come,`` the Walrus said,
.BR
``To talk of many things:
.I .5
.B
Of shoes\(\em and ships\(\em and sealing wax\(\em
.B
Of cabbages\(\em and kings\(\em
.B
And why the sea is boiling hot\(\em
.B
And whether pigs have wings.``
```

The `.Vw` is *another* foil-start macro. We will see exactly what it does in the next section. A bit later on, we will also find out what the other macros in this example (`.T`, `.BR`, and `.I`) are all about.

3. THE MACROS THEMSELVES

The time has come to explain all the MV macros in detail.³

3.1 Foil-Start Macros

Each foil *must* start with a foil-start macro. There are nine foil-start macros for generating nine different-sized foils; the names (and the corresponding mounting-frame sizes) of these macros are shown in Table I.

TABLE I
Foil-Start Macros

Macro Name	Size* and Type	BTL Frame Number†
.VS	7×7 view graph <i>or</i> 2×2 super-slide	E-7351 <i>or</i> E-7351-R
.Vw	7×5 view graph	E-7351-B
.Vh	5×7 view graph	E-7351-A
.VW	9×7 view graph	E-8814 <i>or</i> E-9148
.VH	7×9 view graph	E-8814 <i>or</i> E-9148
.Sw	7×5 35mm slide	E-7351-B
.Sh	5×7 35mm slide	E-7351-A
.SW	9×7 35mm slide	E-8814 <i>or</i> E-9148
.SH	7×9 35mm slide	E-8814 <i>or</i> E-9148

* Size of the mounting frame opening (width times height) in inches.

† BTL stock item number.

The naming convention for these nine macros is that the first character of the name (V or S) distinguishes between view graphs and slides, while the second character indicates whether the foil is square (S), small wide (w), small high (h), big wide (W), or big high (H). Slides are

2. The input string `\(\em` is the *troff* name for the "em dash" (long dash).

3. The MV macros are summarized in *mv(7)*.

“skinnier” than the corresponding view graphs: the ratio of the longer dimension to the shorter one is larger for slides than for view graphs. As a result, slide foils can be used for view graphs, but not vice versa; on the other hand, view graphs can accommodate a bit more text.

Note that .VW and .SW produce foils that are 7×5.4 inches because commonly available typesetter paper is less than 9 inches wide; these foils must be enlarged by a factor of 9/7 before they can be used as 9-inch-wide by 7-inch-high view graphs.

Each foil-start macro causes the previous foil (if any) to be terminated, foil separators to be produced, and certain heading information to be generated. The *default* heading information consists of three lines of *right-justified* data:

- The current date in the form *mo/dy/yr*
- BTL
- Ⓐ FOIL *n*

where *n* is the sequence number in the current “run”; as explained below, this heading information is replaced by the three arguments of the foil-start macro, if those arguments are given.

The actual projection area is marked by “cross-hairs” (plus signs) that fit into the corners of the view graph mount, helping one to position the foil for mounting.

All foils other than the square (.VS) foil also have a set of (horizontal or vertical) “crop marks”; these indicate how much of the foil will be seen if it is made into a slide, rather than into a view graph.

The default heading information can be changed by specifying three optional arguments to the foil-start macro (we use the square brackets [] to indicate that the argument they enclose is optional):

```
.XX [ n ] [ id ] [ date ]
```

where *XX* stands for one of the nine foil-start macros, *n* is the foil identifier (typically a number), *id* is other identifying information (typically the initials of the person creating the foil), and *date* is usually the date. The resulting heading information consists of three lines of right-justified text: *id*, *date*, and FOIL *n*. If *date* and *id* are omitted on a foil-start macro, then the corresponding values (if any) from the previous foil-start macro are used.

See the Appendix for examples of all this.

3.2 Level Macros

The MV macros provide four levels of indentation, called .A, .B, .C, and .D. Each of these *level macros* causes the text that follows it to be placed at the corresponding level of indentation.

3.2.1 The .A Level

The leftmost (left margin) level is obtained by:

```
.A [ x ]
```

The .A level is automatically invoked by each of the foil-start macros. Each .A macro spaces a half-line from the preceding text, unless the *x* argument is specified (*x* can be any character or string of characters); *x* suppresses the spacing.

The .A macro can also be invoked through the .I macro (see §3.4).

3.2.2 The .B Level

```
.B [ mark [ size ] ]
```

The .B level items are marked by a bullet (in a slightly reduced point size). The text that follows the .B macro is spaced one half-line from the preceding text.

The .B level mark may be changed by specifying the desired mark (which may be any character string⁴) as the first argument (*mark*). Without the second argument, the point size of the mark is not reduced. Thus, one can produce a numbered list as follows:

```
.VS
This is a list of things:
.B 1.
This is thing number 1.
.B 2.
This is thing number 2.
.B 3.
This is the third and last thing on this foil.
```

It is possible to change the point size of the mark: the second argument (*size*), if given, specifies the desired point-size change. An unsigned or positive (+) argument is taken as an increment; a negative (-) argument is a decrement; an argument greater than 99 causes the mark to be reduced in size just as if it were the default mark, namely the bullet. After the mark is printed, the previous point size is restored. All these point-size changes are *completely* invisible to the user.

3.2.3 The .C Level

```
.C [ mark [ size ] ]
```

The .C level is just like the .B level *except* that it is indented farther to the right than the .B level and the default mark is a long (*em*) dash (—), also in a slightly reduced point size.

3.2.4 The .D Level

```
.D [ mark [ size ] ]
```

The .D level's default mark is a bullet (smaller than that used for the .B level); the .D level is indented farther to the right than the .C level and it does *not* space from the previous text; it just causes the following text to start on a new line (in other words, it causes a *break*—see §3.10). Otherwise, it behaves just like the .B and .C levels.

3.2.5 More about Levels

- The .A macro never generates a mark of any sort; it is the “left-margin” macro.
- Repeated .A calls are ignored, but each successive call of any of the other three level macros generates the corresponding mark.
- The amount of vertical pre-spacing done by each level macro can be changed with the .DV macro (see §3.7).
- Example 3 in the Appendix is devoted to the level macros.

3.3 Titles

The title macro .T creates a centered title from its argument:

```
.T string
```

The size of the title is four points larger than the prevailing point size. Remember that the argument must be enclosed within double quotes (“...”) if it contains blanks. Any indentation established by the .I macro (see §3.4) has no effect on titles; they are always centered within the foil's horizontal dimension.

See Examples 2, 3, and 4 in the Appendix.

4. All character-string arguments that contain blanks must be quoted (“...”).

3.4 Global Indents

The entire text (except titles) of the foil may be shifted right or left by the `.I` macro:

```
.I [ indent ] [ a [ x ] ]
```

The first argument is the amount of indentation that is to be used to establish a new left margin. This argument may be signed positive or negative, indicating right or left movement from the current margin. If unsigned, the argument specifies the new margin, relative to the *initial* default margin. If the argument is not dimensioned, it is assumed to be in inches (see [3,8] for legal *troff* units). If the argument is null or omitted, `0i` is assumed, causing the margin to revert to the initial default margin.

If a second argument is specified, the `.I` macro calls the `.A` macro (see §3.2.1) before exiting. The third argument, if present, is passed to the `.A` macro.

See Examples 2, 4, 5, and 7 in the Appendix.

3.5 Point Sizes and Line Lengths

Each foil-start macro begins the foil with an appropriate default point size⁵ and line length. The prevailing point size and the line length may be changed by invoking the `.S` macro:

```
.S [ ps ] [ ll ]
```

If *ps* is null, the *previous* point size is restored. If *ps* is signed negative, the point size is decremented by the specified amount. If *ps* is signed positive, it is used as an increment, and if *ps* is unsigned, it is used as the new point size. If *ps* is greater than 99, the *initial default* point size is restored (see Table II). Vertical spacing is always 1.25 times the current point size.

The second argument, if given, specifies the line length. It may be dimensioned. If it is *not* dimensioned and less than 10, it is taken as inches; if it is *not* dimensioned and greater than or equal to 10, it is taken as *troff units* (1/432nds of an inch); see also §7.3.

See Examples 4, 5, and 6 in the Appendix.

3.6 Default Fonts

The macros assume that the Helvetica (also known as Geneva) Regular font, mounted in position 1, is the default font. Additional fonts can be mounted and the default font can be changed:

```
.DF n font [ n font ... ]
```

The `.DF` macro informs *troff* that *font* is in position *n*. The first-named font is the default font. Up to four pairs of arguments may be specified.

The `.DF` macro must immediately precede a foil-start macro; the initial setting is equivalent to:

```
.DF 1 H 2 I 3 B 4 S
```

See Examples 4 and 5 in the Appendix.

3.7 Default Vertical Space

The vertical space macro allows one to change the vertical pre-spacing done by each of the four level macros (see §3.2):

```
.DV [ a ] [ b ] [ c ] [ d ]
```

5. The default point sizes for each type of foil are given in Table II.

The first argument (*a*) is the spacing for `.A`, *b* is for `.B`, *c* is for `.C`, and *d* is for `.D`; all non-null arguments must be dimensioned; null arguments leave the corresponding spacing unaffected; the initial setting is equivalent to:

```
.DV .5v .5v .5v 0v
```

3.8 Underlining

The underline macro `.U` takes one or two arguments:

```
.U string1 [ string2 ]
```

The first argument is the string of characters to be underlined. The second argument, if present, is not underlined, but concatenated to the first argument.

Example:

```
.U phototypesetter
```

produces:

```
phototypesetter
```

while:

```
.U under line
```

produces:

```
underline
```

See also Example 4 in the Appendix.

3.9 Synonyms

The MV macro package also recognizes the following upper-case synonyms for the corresponding lower-case *troff* requests:

```
.AD .BR .CE .FI .HY .NA .NF .NH .NX .SO .SP .TA .TI
```

See [8] for definitions of the corresponding *troff* requests.

3.10 Breaks

The `.S`, `.DF`, `.DV`, and `.U` macros do *not* cause a break; the `.I` macro causes a break *only* if it is invoked with more than one argument; all the other MV macros *always* cause a break. The *troff* synonyms (see §3.9) `.AD`, `.BR`, `.CE`, `.FI`, `.NA`, `.NF`, `.SP`, and `.TI` also cause a break.

3.11 Text Filling, Adjusting, and Hyphenation

By default, the MV macros fill, but neither adjust nor hyphenate text. This is an aesthetic judgement that seems correct for foils. These defaults can, of course, be changed by using the `.AD`, `.FI`, `.HY`, `.NA`, `.NF`, and `.NH` macros (see §3.9).

4. THE TROFF PREPROCESSORS

It is possible to use the various *troff* preprocessors to typeset foils that require more powerful formatting capabilities.

4.1 TBL for tables

The `tbl(1)` program can be used to set up columns of data within a view graph or slide. The `.TS` and `.TE` macros are *not* defined in the MV macro package, but are merely flags to `tbl(1)`; see [5], as well as Examples 4 and 7 in the Appendix.

4.2 EQN for Mathematical Expressions

The *eqn(1)* program can be used to typeset mathematical expressions and formulas on foils, *provided* one is careful to specify the proper fonts and point sizes; see [4], as well as Examples 5 and 7 in the Appendix. The *.EQ* and *.EN* macros are *not* defined in the MV macro package:

4.3 CW for Constant-Width Program Examples

The constant-width font simulates computer-terminal and line-printer output, and can be sometimes effective in presenting computer-related topics; see *cw(1)*, as well as Examples 5 and 6 in the Appendix.

5. THE FINISHED PRODUCT

5.1 Phototypesetter Output

The typeset output is obtained via the command:

```
mvt [ options ] file_name ...
```

where *file_name* contains the text and the macro invocations for the foils, and *options* can be one or more of the following:

- a preview output on a terminal (other than a Tektronix 4014—see §5.2)
- e invoke *eqn(1)*
- t invoke *tbl(1)*
- Tterm* direct output to *term*, where *term* can be one of the following:

st	STARE
4014	Tektronix 4014
vp	Versatec printer

Using a hyphen (-) in place of *file_name* causes the *mvt* command to read the standard input (rather than a file), as in the following example using the *cw(1)* preprocessor (see §4.3):

```
cw [ options ] file_name ... | mvt [ options ] -
```

5.2 Output Approximation on a Terminal

One can obtain an approximation of the typeset output by entering the command:

```
mvt -a file_name ...
```

The resulting output shows the formatted foils except that:

- Point-size changes are not visible.
- Font changes cannot be seen.
- Titles that are too long appear proper.
- All horizontal motions are reduced to one horizontal space to the right.
- All vertical motions are reduced to one vertical space down.

Thus, for example, it appears that the lines of text following a *.B*, *.C*, or *.D* macro do not align properly (even though, in fact, they do).

Although alignment cannot be determined from this approximation, one can observe line breaks and the amount of vertical space used by the text. If the foil is not full, the macro package prints the number of blank lines (in the then current point size) that remain on the foil; if the foil is full, a warning is printed. If the text *did* overflow the foil, text will be printed after the “cross-hairs.”

5.3 Making Actual View Graphs and Slides

The output of the typesetter is so-called "mechanical paper," which is white, opaque photographic paper with black letters. There are several very simple processes (e.g., Thermofax, Bruning) for making transparent foils from opaque paper. Because some of these processes involve heat, and because mechanical paper is heat sensitive, one should first make copies of the typesetter output on a good-quality office copier, and then use these copies for making the transparencies.

Getting slides made is a much more complicated photographic process that is best left to professionals. It is possible to make both positive (opaque letters on transparent background) and negative (transparent letters on opaque background) slides, as well as colored-background slides, etc.

6. OPINIONS AND SUGGESTIONS

The following suggestions and authors' opinions have been derived from experience, from the examination of several other macro packages for making foils [6,7,10,11,12], and from some publications that discuss good and bad foil-making practices [1,9]:

☞ Foils 3, 4, and 7 in the Appendix violate some or most of the "rules" given below.

- The most useful foil sizes are .VS and .Vw (or .Sw). This is because most projection screens are either square, or "wide" (i.e., are wider than they are tall), and also because the resulting foils are smaller, easier to carry, and require no enlargement before use.
- The *default* point size for each type of foil (see Table II) is the *smallest* point size that will result in a foil that is legible by an audience of more than a dozen people. Reducing the point size below the default should be avoided. If you have more text than fits onto a foil, *don't* reduce the point size; use two or more foils instead.
- Don't abuse font changes. Most novice foil-makers tend to use too many typefaces, resulting in foils that look cluttered and distract the viewer. Be *sure* you need to use a different typeface (e.g., for special emphasis); when in doubt, stick to a single typeface. You should almost never use more than two different typefaces on a single foil.
- Even though this package contains a macro for underlining, don't use it: underlined typeset text almost always looks awful; instead, if necessary, use a different typeface.
- The Helvetica sans-serif font (which is the default font used by this package) is "fatter" and is, *in foils*, considered easier to read than, for instance, the Times Roman serif font used for typesetting normal, "running" text (such as the text in this paragraph). On the other hand, the Times Roman font will allow you to "squeeze" a bit more text onto a foil. If you intend to use italic and/or bold typefaces in your foils, you probably want to mount (via the .DF macro—see §3.6) either the Helvetica Regular, Italic, and Medium:⁶

```
.DF 1 H 2 HI 3 HM
```

or the Times Roman regular, italic, and bold:

```
.DF 1 R 2 I 3 B
```

Bold typefaces tend to be a bit overwhelming. On the whole, the choice of fonts is primarily a matter of personal aesthetics. The examples in the Appendix use the following fonts:

6. Helvetica Medium is really a bold typeface.

1, 2, and 3: H (default)
 4 and 7: R and I
 5 and 6: R and CW

- If possible, use the `.SP` macro to insert a bit of additional white space (say, `.5v` or `1v`, where `v` means “vertical space”) at the top of each foil (i.e., increase the top margin).
- Some people believe that foils should not contain any lower-case alphabetic characters to maximize legibility; to the best of our knowledge, this is exactly the opposite of the truth: “normal” upper-and-lower-case text is far more legible than upper-case-only text.⁷ Upper-and-lower-case alphabets have evolved and been refined over millenia precisely because they result in more legible text. Furthermore, such text is less “bulky” than upper-case-only text, so one can get more information onto a foil without crowding it.
- Make all the foils for a presentation as consistent as possible; changing fonts, typefaces, point sizes, etc., from foil to foil tends to jar and distract the viewer. While it is possible to introduce emphasis and draw the viewer’s attention to particular items with such changes, this only works if you do it very purposefully and very sparingly; overuse of these techniques is almost always counter-productive.

In summary, the dictum that “the medium is the message” doesn’t apply to foil making; so when in doubt:

- Don’t change point sizes.
- Don’t change fonts or typefaces.
- Don’t underline!
- Use many “sparse” foils, rather than few “dense” ones.
- Use fewer words, rather than more.
- Use larger point sizes, rather than smaller.
- Use larger top and bottom margins, rather than smaller.
- Use normal upper-and-lower-case text, rather than upper-case only.

7. WARNINGS

7.1 Use of Troff Requests

In general, it is not advisable to intermix arbitrary *troff* requests with the MV macros, because this often leads to undesirable (and sometimes downright astonishing) results. The “safe” requests are the ones for which upper-case synonyms have been defined in the MV package (see §3.9). Other *troff* requests should be used sparingly (if at all), and with care and discipline. Particularly dangerous are the requests that affect point size, indentation, page offset, line and title lengths, and vertical spacing between lines. Use the `.S` and `.I` macros instead (see §3.5 and §3.4).

7.2 Reserved Names

Certain names are used internally by this macro package. In particular, all two-character names starting with either `)` or `]` are reserved. Names that are the same as names of the MV macros and strings described in this manual, or the same as any *troff* names [8], cannot be used either. Furthermore, if any of the preprocessors (see §4) are used, their reserved names must also be avoided.

7. The only exception to this rule are foils set in a point size so small that lower-case characters simply can’t be read; this is usually the case for foils produced on a normal typewriter.

7.3 Miscellaneous

The .S macro changes the point size and vertical spacing immediately, but a line-length change requested with that macro does not take effect until the next level-macro call.

Specifying a *third* argument to the .S macro usually results in a disaster.

The string Tm (invoked as *(Tm) generates a trademark symbol.

The tilde (~) is defined by the MV macros as a "non-paddable" space; that is, the tilde may be used wherever a fixed-size (non-adjustable) space is desired. To override this definition, include the following line in your input file:

```
.tr --
```

8. DIMENSIONAL DETAILS

Table II shows, for each style of view graph, the default point size, the maximum number of lines of text (at the default point size), and the height, width, and "aspect ratio," both nominal and actual.

TABLE II
Default Point Size, Dimensions, and Aspect Ratios

Macro Name	Point Size	Max. Lines	Nominal				Actual (Text)			
			Width (inches)	Height (inches)	Aspect Ratio	$\frac{1}{AR}$	Width (inches)	Height (inches)	Aspect Ratio	$\frac{1}{AR}$
.VS	18	21	7	7	1	1	6	6.8	1.13	.88
.Vw	14	19	7	5	.71	1.4	6	4.8	.8	1.25
.Vh	14	27	5	7	1.4	.71	4.2	6.8	1.6	.62
.VW	14	21	7	5.4	.77	1.3	6	5.2	.87	1.15
.VH	18	28	7	9	1.3	.77	6	8.8	1.5	.68
.Sw	14	18	7	4.6	.67	1.5	6	4.4	.73	1.4
.Sh	14	27	4.6	7	1.5	.67	3.8	6.8	1.8	.56
.SW	14	18	7	4.6	.67	1.5	6	4.4	.73	1.4
.SH	18	28	6	9	1.5	.67	5	8.8	1.76	.57

NOTES: "Max. Lines" is the maximum number of lines of text at the *default* point size.

"Aspect Ratio" (AR) is the ratio of height over width.

Remember that, normally, each .A, .B, and .C macro generates a 1/2-line space.

The .SW (if used as a view graph) and .VW foils must be enlarged by a factor of 9/7.

ACKNOWLEDGEMENTS

We thank the many users of MV who provided the feedback necessary to refine the various features of this package in the early stages of its development and who were willing to use it despite the fact that, during the first several years of its existence, the only available user "documentation" was by word of mouth.

REFERENCES

- [1] Bell Laboratories. *Visual Aid Standards* (1973—out of print).
- [2] Dolotta, T. A., Olsson, S. B., and Petrucelli, A. G. (eds.). *UNIX User's Manual—Release 3.0*, Bell Laboratories (June 1980).
- [3] Kernighan, B. W. *A TROFF Tutorial*, Bell Laboratories.
- [4] Kernighan, B. W., and Cherry, L. L. *Typesetting Mathematics—User's Guide* (Second Edition), Bell Laboratories.
- [5] Lesk, M. *TBL—A Program to Format Tables*, Bell Laboratories.
- [6] McGill, R. *VMAC—Commands for Preparing Vu-graphs or Posters*, Bell Laboratories (1976).
- [7] Noll, J. C. *ATS Bulletin 77-5*, Bell Laboratories (1977).
- [8] Ossanna, J. F. *NROFF/TROFF User's Manual*, Bell Laboratories.
- [9] Perry, R. E. *Audience Requirements for Technical Speakers*, American Federation of Information Processing Societies (1971).
- [10] Renkel, W. H. *VGEL—View Graph Extended Language*, Bell Laboratories (1978).
- [11] Sturman, J. N. *MVIEW—A Set of Macrocommands for the Generation of View Graphs*, Bell Laboratories (1978).
- [12] Vogel, G. C. *Easy Phototypeset View Graphs*, Bell Laboratories (1977).

APPENDIX

This Appendix contains seven examples. The input for Examples 1 and 2 is given in §2 above. The input for each of the other examples precedes the corresponding view graph. Note that the *output* of Example 6 is, essentially, the *input* for Example 7.

EXAMPLE 1:

9/15/81
BTL
⊗ FOIL 1

Six stages of a project:

- wild enthusiasm
- disillusionment
- total confusion
- search for the guilty
- punishment of the innocent
- promotion of the non-participants

EXAMPLE 2:

June 29, 1980
Less Trivial
⊗ FOIL 2

What the Walrus Said

"The time has come," the Walrus said,
"To talk of many things:

- Of shoes—and ships—and sealing wax—
- Of cabbages—and kings—
- And why the sea is boiling hot—
- And whether pigs have wings."

EXAMPLE 3:

```

.Vh 3 "Levels & Marks"
.T "Foil Levels & Level Marks"
This is the .A (left margin) level;
.B
this is the .B level,
.B
as is this;
.C
this is the .C level,
.C
as is this;
.D
and this is the .D level,
.D
as is this.
.A
The large bullet, the dash, and the small
bullet are the default ``marks`` for
levels .B, .C, and .D, respectively.
However, these three levels can also
be marked arbitrarily:
.B B.
Like this (this is the .B level);
.C 3.
like this (this is the .C level);
.D d.
like this (this is the .D level), or
.D iv.
like this, or even
.D \(\rh-\(\bu +4
like this.
.A
The .A level cannot be marked.
.B
An arbitrary number of lines of text
can be included in any item at any level;
the text will be filled, but neither adjusted
nor hyphenated, just like this .B level item.

```

Foil Levels & Level Marks

This is the .A (left margin) level;

- this is the .B level,
- as is this;
 - this is the .C level,
 - as is this;
 - and this is the .D level,
 - as is this.

The large bullet, the dash, and the small bullet are the default "marks" for levels .B, .C, and .D, respectively. However, these three levels can also be marked arbitrarily:

B. Like this (this is the .B level);

3. like this (this is the .C level);

d. like this (this is the .D level), or

iv. like this, or even

☞ ● like this.

The .A level cannot be marked.

- An arbitrary number of lines of text can be included in any item at any level; the text will be filled, but neither adjusted nor hyphenated, just like this .B level item.

EXAMPLE 4:

```

.DF 1 R
.VS 4 Complex
.T "Of Bits & Bytes & Words"
.S -4
.I 3 A x
.ft I
But let your communication be, Yea, yea;
Nay, nay: for whatsoever is more than these
cometh of evil.*
.ft
.I +1 a nospace
Matthew 5:37
.BR
.S
.I 0 .A
Binary notation has been around for a
.S +6
long
.S
time.
.B
The above verse tells us to use:
.C 1)
binary notation,
.ft I
and
.ft
.C 2)
redundancy
.D \(rh
(in communicating)
.B
Binary notation is
.U not
suited for human use, above verse to
the contrary notwithstanding.
.SP
.S -2
.TS
box ;
c | c | c | c
l | c | c | c .
System Bits/Byte      Bytes/Word      Bits/Word
IBM-7090/94      6      6      36
IBM-360/370      8      4      32
PDP-11/70        8      2      16
.ft
.S
.S -4
.U -----
.BR
* The use of this verse in this context
is plagiarized from C. Shannon.
.S

```


June 29, 1980
Complex
Ⓢ FOIL 4

Of Bits & Bytes & Words

*But let your communication be,
Yea, yea; Nay, nay: for whatsoever
is more than these cometh of evil.*
Matthew 5:37*

Binary notation has been around for a **long** time.

- The above verse tells us to use:
 - 1) binary notation, *and*
 - 2) redundancy
☞ (in communicating)
- Binary notation is not suited for human use, above verse to the contrary notwithstanding.

System	Bits/Byte	Bytes/Word	Bits/Word
IBM 7090/94	6	6	36
IBM 360/370	8	4	32
PDP 11/70	8	2	16

* The use of this verse in this context is plagiarized from C. Shannon.

EXAMPLE 5:

```

.de CW
.I .5 a
.NF
..
.de CN
.FI
.I 0 a
..
.DF 1 R 2 I 3 CW
.VS 5 "CW & EQN"
.EQ
gsize 18
.EN
.S 100 5.5
Input:
.CW
.EQ
sum from k=1 to inf m sup k-1
~-- 1 over 1-m
.EN
.CN
Output:
.I 2 a
.EQ
sum from k=1 to inf m sup k-1
-- 1 over 1-m
.EN
.I 0 a
Input:
.CW
The equation $ f(t) ~-- 2 pi
int sin ( omega t ) dt $
is used here in running text,
rather than being displayed.
.CN
Output:
.I .5 a
.EQ
delim $$
.EN
.AD
The equation $ f(t) ~-- 2 pi
int sin ( omega t ) dt $
is used here in running text,
rather than being displayed.
.EQ
delim off
gsize 10
.EN

```

Input:

```
.EQ
sum from k=1 to inf m sup k-1
  ~ ~ 1 over 1-m
.EN
```

Output:

$$\sum_{k=1}^{\infty} m^{k-1} = \frac{1}{1-m}$$

Input:

The equation \$ f(t) ~ ~ 2 pi
int sin (omega t) dt \$
is used here in running text,
rather than being displayed.

Output:

The equation $f(t) = 2\pi \int \sin(\omega t) dt$ is
used here in running text, rather than
being displayed.

EXAMPLE 6:

```

.VS 6 "The Works: Input"
Input:
.S -4
.CW
.TS
center doublebox ;
Cip+4 | Cip+4 S S
^ | L L L
^ | C | C | C
^ | C | C | C
Li | C | C | N .
Users\(->Hardware
\(->_ \(->_ \(->_
\(->UNIX\*(Tm\(->Model\(->Serial
\(->System\(->\^ \(->Number
=
OS Dev.\(->A\(->VAX\(->54
SGS Dev.\(->B\(->11/70\(->3275
Low-End\(->C\(->11/23\(->221

And now ... \(->T{
.NA
Some filled text and an equation:
T)\(->T{
$ zeta (s) = prod
from k=1 to inf k sup -s $
.AD
T)\(->1.2
.TE
.CN

```

(\(-> = tab)

+

+

Input:

```

.TS                                     (→ = tab)
center doublebox ;
Cip+4 | Cip+4 S S
^ | L L L
^ | C | C | C
^ | C | C | C
Li | C | C | N .
Users→Hardware
→_→_→_
→UNIX\*(Tm→Model→Serial
→System→\^→Number
=
OS Dev.→A→VAX→54
SGS Dev.→B→11/70→3275
Low-End→C→11/23→221

-
And now ...→T{
.NA
Some filled text and an equation:
T}→T{
$ zeta (s) = prod
from k=1 to inf k sup -s $
.AD
T}→1.2
.TE

```

+

+

EXAMPLE 7:

```

.VS 7 "The Works: Output"
.EQ
delim $$
gsize 14
.EN
Output:
.I 0 a
.SP
.TS
center doublebox ;
Cip+4 | Cip+4 S S
^ | L L L
^ | C | C | C
^ | C | C | C
Li | C | C | N .
Users Hardware

      UNIX\*(Tm      Model  Serial
      System \^      Number

=
OS Dev.      A      VAX      54
SGS Dev.     B      11/70   3275
Low-End      C      11/23   221

And now ...  T{
.NA
Some filled text and an equation:
T}          T{
$ zeta (s) = prod
from k=1 to inf k sup -s $
.AD
T}          1.2
.TE
.EQ
delim off
gsize 10
.EN

```

Output:

<i>Users</i>	<i>Hardware</i>		
	UNIX™ System	Model	Serial Number
<i>OS Dev.</i>	A	VAX	54
<i>SGS Dev.</i>	B	11/70	3275
<i>Low-End</i>	C	11/23	221
<i>And now ...</i>	Some filled text and an equation:	$\zeta(s) = \prod_{k=1}^{\infty} k^{-s}$	1.2