Braille2000, LLC

6801 Southfork Cir, Lincoln, NE 68516
Phone: (402) 423-4782, Fax: (402) 423-5154
http://www.braille2000.com

A Guide to

Braille2000 V2

Overview

Braille2000 Version 2 (V2) is a more powerful, more modern Braille2000 than Braille2000 Version1 (V1). This document presents the major extensions and differences and also describes in detail the entire user interface for V2.

Braille2000 V2 is compatible with Windows 8 and Windows 10 (as well as older Windows).

Braille2000 V1 and V2 can operate at the same time, but not using the same Internet license.

Major Differences (relative to Braille2000 V1)

1. Different filename suffix (.b2k)
2. Up/Down-arrow into yellow and gray
3. Different Speedbraille™ macros
4. No aqua-tint to the Print View (aqua-tint now indicates "block" material (lists and tables))
5. Different procedure to import RTF file, revised/extended translation typing directives (now called %directives)
6. Red triangles are now circles and can be clicked to highlight whole paragraphs
7. Braille Page Break is now handled by Page Control
8. Support for adjusted margins (left and right) in Layout Control
9. Dropping of support for V2 on Windows95, Windows98, Windows-ME
10. There are two cursor positions between adjacent cells

Installing Braille2000 V2

If you are already using V2, click the File Management button and select Fetch Software Update. Click the drop-down button for the Component list and select BRL2000B 2.275.

You can also get the setup file BRL2000B_Setup.Exe from www.braille2000.com ("Download the latest release…" and then click the Download button). If your browser offers to let you "Run" the download, do that. Else, save the file to a convenient location (the Desktop is handy) and after the download is complete, launch the file to run it. You may need to turn off your anti-virus system during the installation. You **will** need to have administrator capabilities (use a Windows logon that is at the Administrator level).

User Interface

Display

There are four display modes: A (ASCII), B (braille, i.e., simbraille), P (print), and S (source). These can also be called the ASCII View, Braille View, Print View, and Source View, respectively. The mode can be selected via four adjacent buttons in the toolbar. The ASCII and Braille views present the braille page, using Roman characters (so called ASCII-braille), or simbraille, respectively. The Print and Source views present print text. The Print view retains the same size and shape as the braille page, with print text positioned to match its braille equivalent. When the print equivalent and/or auxiliary material (such as an embedded image from a NIMAS file) would cause the layout to deviate from that of the braille page, the auxiliary material is omitted and/or the tall math expression is replaced with its equivalent in simbraille. The Source view allows the page layout to deviate (usually in the vertical direction) from the braille page, thereby allowing tall math expressions and auxiliary material (such as images) to show in place. The Source view thereby gives a better print depiction of the material while the Print view allows the viewer to see the layout of the resulting braille page.

The default background color scheme is as follows:

Color	Usage
White	normal prose
Lt. Blue	special text blocks (e.g., table, list, etc.)
Yellow	generated page data (running head, page numbers etc.)
Gray	fill (empty voids as well as automatic format-required punctuation)
Red	warnings (spelling-suspect words)

Note that the light cyan tint of the V1 Print View is no longer the default background color for normal prose in the V2 Print View (white is used in all viewing modes for normal prose).

The first line of each paragraph is marked with a colored circle in the left border of the editing panel. The color changes to blue when the line is also the first line in a block (table, nested list, note, drawing). The interior of the circle becomes colored when the entire paragraph is selected (e.g., highlighted). Clicking the circle with the mouse selects the entire paragraph. Blocks are shown with a powder-blue background tint (to remind the user that editing behaviors are specialized in such areas of the file, e.g., the use of the Tab key within a table).

Keyboard and Cursor

Navigation

In V2, the blinking cursor can exist within yellow and gray areas. As before you cannot directly enter keyboard data into such regions.

The up- and down-arrow keys (and the mouse) can move the cursor to any line regardless of color, however the left- and right-arrow keys can move the cursor only to editable positions, i.e., not into yellow or gray. If the cursor is already within yellow or gray (via the up- or down-arrow or the mouse), either left/right arrow key moves to beyond the yellow or gray.

By default, the leading and trailing blanks on each line are jumped over when using left- or right-arrow. This is controlled by "Arrow keys skip margins" in the dialog box reached via Adjust and Keyboard. Regardless of that setting, you can arrow-key into a margin using Alt-left-arrow or Alt-right-arrow.

Document control points (annotations) may exist between cells (between characters in the print view) and can be visualized by small yellow triangles (use Adjust / Display to control whether or not they show). The left- and right-arrow keys move the cursor cell by cell. But if you look carefully, you will see that there are actually two possible cursor positions between the cells, one is a hair to the left of any annotation and the other is a hair to the right of any annotation. The left- and right-arrow keys do not provide for any hair-width cursor movements (you can do that via the menu using Menu / Edit / Insert / "Flip gap bias" to move the cursor from one side of annotations to the other, at the same cell gap). But you can indirectly cause that effect using left- and right-arrow keys, because left-to-right (cell-to-cell) movement by the right-arrow key lands the cursor on the left side of any annotations, while right-to-left (cell-to-cell) movement by the left-arrow key lands the cursor on the right side of any annotations. Thus right- and then left-arrow places the cursor to the right of annotations while left- and then right-arrow places the cursor to the left of annotations.

The two inter-cell (gap) cursor positions can be distinguished visually. Here is an enlargement of the cursor positioned *before* vs. *after* the annotations. Keyboard entry and Paste functions occur *after* annotations when the cursor is to the right, but *before* them when the cursor is to the left. If the annotation is, say, a bookmark, this positioning determines whether new material is before vs. after the bookmark. If the annotation is, say, a Code change, this positioning determines whether the new material is in the first code versus the second code.

When used with the Ctrl key the arrow keys behave as follows:

Key	Effect
Ctrl-left-arrow	jumps left to the next word beginning
Ctrl-right-arrow	jumps right to the next work ending
Ctrl-up-arrow	moves up one line and scrolls down one line*
Ctrl-down-arrow	moves down one line and scrolls up one line*

*Page View Scrolling is turned "off"

The Home and End keys jump the cursor to useful locations, as per the following table:

Key	Effect
Home	first non-blank position in the same line
Home x2	first position in the same line
Home x3	first non-blank position in the same paragraph
Home x4	first non-blank position in the same page
Home x5	first non-blank position in the file
Ctrl-Home	first non-blank position in the file
End	after last non-blank position in the same line
End x2	after last position in the same line
End x3	after last non-blank position in the same paragraph
End x4	after last non-blank position in the same page
End x5	after last non-blank position in the file
Ctrl-End	after last non-blank position in the file

The notation xN (as in x2) denotes the number of times the key is pressed in succession without any other keypress intervening. Thus if you press Home three times in a row the cursor jumps to the start of the paragraph.

The Pg Up and Pg Down keys move the cursor up or down by the number of lines showing on the screen (usually 25). When Page View Scrolling is "on" and the screen shows at least the number of lines as on a page, the cursor moves exactly one page regardless of the number of lines actually showing. When Pg Up or Pg Down is pressed with Ctrl then Page View Scrolling is turned "on" (unless the screen shows fewer lines than one page) and the cursor moves to the first non-blank position on the page (as if Home x4 were implied).

Page-View Scrolling

There are two scrolling modes: line scrolling and page scrolling (called Page-View). In page scrolling the display scrolls by whole pages, but this mode is available only when the editing panel shows at least a whole page at a time. The current page scrolling mode is indicated by the appearance of the "pv" button in the extreme lower-right corner of the window (where the vertical and horizontal scroll bars meet). The legend ("pv") is dark when page scrolling is active and grayed out when line scrolling is active. Assuming the screen shows a whole page, the status can be toggled by clicking the "pv" button with the mouse. The setting is also changed by using Ctrl-Up/Down-Arrow and Ctrl-Pg-Up/Down.

Editing

The following keys are used to change the content of the document:

Key	Effect
symbol keys	In typewriter input mode, the symbol keys insert or overtype the position where the cursor is located. If the cursor is within a yellow or gray area, it first jumps out of yellow and/or gray moving the same direction (forward or reverse) that it moved when entering the yellow or gray area (mouse click cursor positioning implies forward momentum). In braillewriter input mode, the keys sdfjkl form braille cells as if a braille typewriter. Other letters trigger SpeedBraille™ Keys. The digits 0 to 9 (shifted or unshifted) input the corresponding ASCII-braille cell unless a SpeedBraille definition is given for that key.
Enter	When the cursor is not within a table, the enter key inserts a paragraph break. When used with Shift, it inserts a line break. Within a table, the enter key moves the cursor to the first column of the next row or if the cursor is in the last row, a new row is created and the cursor jumps to the first column of that row.

Backspace	Without the Ctrl key, the backspace key moves the cursor like left-arrow and then deletes one character or cell (as per Delete, below) except that when moving across a non-deleteable boundary (a column separation in a table) no delete action is performed. With the Ctrl key, the backspace key merges the current paragraph with the preceding one, leaving one space between the two.
Delete	Without the Ctrl key, the delete key deletes the character to the right of the cursor without moving. This applies also to delete-able boundaries (paragraph boundaries and tabs within tabbed text). If the cursor is to the immediate left of a non-delete-able boundary (a column separation in a table), the delete key merely moves the cursor as per right-arrow. With the Ctrl key, the delete key merges the current paragraph with the following one, leaving one space between the two.
Tab	When the cursor is not within a table, the tab key inserts a tab stop. Within a table, the tab key jumps to the next table column or if the cursor is in the last column, a new table row is created and the cursor jumps to the first column in the new row. Within a table, Shift-Tab moves the cursor to the previous column and if the cursor is already in column 1 it moves the cursor to the last column of the previous row (if already in the first row, the cursor remains in the first row). When the cursor is not within a table, Ctrl-Tab increases the style level (e.g., increases indent level) and Shift-Ctrl-Tab decreases style level.

Highlighting

In Version 2, a highlight is one kind of *Selection*. A selection is a dynamic indicator of one or more selected phrases in the document, such that you can see and/or manipulate those selected phrases all at the same time. The "highlighting" is merely the selection you create via the techniques listed below. Elements highlighted can be transferred to a named selection and stored with the file for later use. A selection can also be established via "Find All" and via document structure patterns, such as by using the NIMAS "dtbook" panel. A file can have zero or more named selections.

Unlike Version 1, selections (including highlighting) cover only prose and not margins (implied blanks usually at the extreme right or left of a line) nor generated text (yellow material) nor void fill (gray material). This difference has a few subtle effects that are different, such as the implied

behavior when replacing a selected (e.g., highlighted) phrase that spans the indent or runover margin area of a paragraph (such areas, containing virtual blank cells, could be highlighted in V1 but not in V2, with the corresponding ability for V1 to make changes to indent or runover settings (not always intended by the user) when manipulating such a selection).

Named and stored selections adjust dynamically to editing and can propagate through cut/copy/paste operations. Named and stored selections can have a "Post-it" note attached to each selected phrase, for example, to convey proofreader comments back to the transcriber.

> The "Information Card" concept from Version 1 has been retained—it is a whole-document note whereas the named selection can give a note per selected phrase.

How to highlight…
By using the mouse

Regular linear highlight

Text can be highlighted by dragging the mouse across the region to be highlighted (position the mouse at the start of the highlight, press and hold the left mouse button while moving the mouse to the end of the highlight, release the mouse button). During mouse action, the start location is fixed and does not move but the end location follows the mouse so long as the left mouse button remains pressed. After the highlight is defined, both ends of the highlight can be adjusted further by holding down the Shift key while you re-drag the mouse to re-establish the start or end location.

Multi-highlight

If you want to create a multi-highlight (to highlight multiple non-contiguous phrases), first form the initial highlight as described above. Highlight a subsequent non-contiguous phrase by holding down the Ctrl key as you start dragging the mouse for the next phrase. Once the highlighting for the new phrase is established, you can use all of the techniques given above to adjust the end-points of the most recently created highlighted phrase. This entire process can be repeated to add additional non-contiguous phrases to the highlighting.

Block highlight

You can create a block highlight (only in the Braille View or ASCII View) by holding down the Alt key as you begin dragging the mouse. When you start a block highlight, any previous highlighting is cleared: you cannot form a multi-block highlight.

By using the keyboard

Regular linear highlight

Text can be highlighted by holding down the Shift key while using any of the cursor positioning keys: the material from the original cursor position to the final position will become highlighted. To discard the highlighting, press Esc.

If "Persistent highlights" (see below) is set to "off", while the cursor remains at the end of a highlighted phrase you can hold down the Shift key and re-adjust the highlight using arrow keys.

If "Persistent highlights" is set to "on", you can use arrow keys to move the cursor to either end of the highlight and then hold down Shift and re-adjust that end by using arrow keys.

Multi-highlight

To create a multi-highlight using the keyboard, "Persistent highlights" (in Keyboard options, reached via Adjust and Keyboard) must be set to "on". In that setting, the arrow keys (by themselves) do not clear the highlight (you can clear the highlight using the ESC key). Highlight the first phrase using the Shift key as described above. Then move the cursor to a new location using just the arrow keys. Then create another highlighted phrase again using the Shift key. If the cursor is right at the end-point of the last-highlighted phrase, using the Shift key with arrow keys adjusts that end-point rather than begin a new highlighted phrase.

Block highlight

To form a block highlight using the keyboard, hold down the Shift and Alt keys while you use arrow keys to expand the size of the block. Begin using the left or right arrow keys to establish an initial breadth to the block highlight (otherwise nothing shows on the screen). Only arrow keys work to form a block highlight (not PgUp or PgDn that work to form a linear highlight).

Note that the block highlight is a braille concept (a block of braille cells with straight sides), and can only be created in the Braille View or the ASCII View.

Via mouse clicks

You can highlight units of prose via multiple left-clicks with the mouse:

Number of clicks	Effect
one	The cursor moves to the mouse pointer
two	The word at the mouse pointer is highlighted
three	The line at the mouse pointer is highlighted
four	The paragraph at the mouse pointer is highlighted
five	The page at the mouse pointer is highlighted
six	The entire file is highlighted

Multiple clicks must occur with no more than a half-second between clicks (i.e., at a moderate pace). If the Ctrl key is pressed during the mouse clicks, the new area is added to the current highlight as a separate phrase (else the current highlighting, if any, is removed and the new area becomes the only area highlighted).

You can also click in the screen region immediately to the left of cell 1, where you see paragraph circles and optional page-line numbers. Clicking on a paragraph circle highlights the whole paragraph. Clicking on a line number or nearby but not on a paragraph circle highlights just that line. Having begun by clicking a line number or a paragraph circle, you can drag the mouse up or down to include more lines/paragraphs.

Changing emphasis…

After highlighting some text (braille or print view, single- or multi-highlight), you can click any of the four emphasis buttons (each a toggle) to adjust the emphasis (typeform) of the highlighted material. The buttons are for Bold, Italics, Script, and Underlined.

Additional emphasis (typeform) control is available via the menu at (Menu / Adjust / Emphasis).

Cut / Copy / Paste

Selected text can be cut or copied to the Clipboard. Both operations put a copy of the selected text on the Clipboard. Only the cut operation then deletes the selected text from the document.

When the selection has multiple phrases, they all arrive at the Clipboard as separate paragraphs. By using special cut or special copy (via right-click to the corresponding button) you can target the cut or copied text to a new workarea rather than to the Clipboard itself.

Text is managed by various kinds of "marks" such as "paragraph break" and sometimes "line break". This discussion is about paragraph marks.

When a phrase is copied to the Clipboard, it has optional beginning and ending ("exterior") marks (invisible to the user) that will be written here as <p>. A phrase can also have interior marks. It is the exterior marks that determine how the phrase joins with other text when pasted. A copied phrase can thus be represented as either of the following. (Interior marks are not mentioned, as if none are present.)

Abstract notation	Behavior
phrase	Paste without adding paragraph breaks
<p>phrase<p>	Paste with leading/following paragraph breaks

The exterior <p> mark is part of the information copied to the Clipboard. **It is indicated or not by the manner of highlighting**: when one or more whole "mark units" (e.g., paragraphs) are highlighted, exterior marks are included; when other than whole units are highlighted, exterior marks are omitted. Exterior marks for paragraphs can also be added by using special paste. When a whole paragraph is highlighted, the paragraph's "paragraph circle" (the beginning of paragraph visual on-screen indicator) becomes colored-in. The marks being discussed here manage the creation of new paragraphs during paste, as follows:

Cursor location for Paste	Effect (based on Nx/Ni numbers of exterior/interior <p> boundary marks pasted)
At the start of a target paragraph	Creates $N_i+N_x/2$ number of new paragraphs
In the middle of a target paragraph	Creates N_i+N_x number of new paragraphs
At the end of a target paragraph	Creates $N_i+N_x/2$ number of new paragraphs

Note: in this scheme leading/trailing <p> marks always go together (both or none), i.e., Nx is always 0 or 2.

Following a Paste operation, the cursor is situated at the end of the added material. This behavior can be changed: right-click the Paste button and select "Cursor afterwards" and then choose the behavior you want, it can be "at End" (the default), "at Beginning", or "Highlight" (the pasted material becomes highlighted).

Keyboard actions (shortcuts)

The Ctrl key used with certain letters will trigger standard menu actions, as follows:
** denotes actions unique to Braille2000 (otherwise the action is common throughout Windows)

Ctrl+a	Select (highlight) all text
Ctrl+b	Bold (toggle)
Ctrl+c	Copy (ordinary copy, not for copy to new workarea)

Ctrl+d	
Ctrl+e**	Emboss
Ctrl+f	Find
Ctrl+g**	Use alternate interpretation
Ctrl+h	
Ctrl+i	Italics (toggle)
Ctrl+j**	Jump to cell
Ctrl+k**	Insert one braille cell (used when keyboard is in typewriter mode)
Ctrl+l	
Ctrl+m	
Ctrl+n**	Find next, Replace next (next in general)
Ctrl+o	Open file
Ctrl+p	Print
Ctrl+q**	Query (when speech option is active)
Ctrl+r	Replace
Ctrl+s	Save file
Ctrl+t**	Display tab ruler
Ctrl+u**	Underline emphasis (toggle)
Ctrl+v	Paste
Ctrl+w**	Toggle between last two view modes
Ctrl+x	Cut
Ctrl+y**	Insert last cell again
Ctrl+z	Undo
Ctrl+F3	Previous view (toggle View Selector)
Ctrl+F4	Close file
Alt+F4	Close application (e.g., close Braille2000)
Ctrl+spacebar	Insert non+breaking space
Ctrl+hyphen**	Insert breaking hyphen
Shift+Ctrl+hyphen**	Insert vanishing hyphen
Ctrl+tab	Increase style level (e.g., list level 2 from list level 1)
Shift+Ctrl+tab	Decrease style level (e.g., list level 1 from list level 2)
F1	Help for current menu item or dialog box
Enter	Start a new paragraph
Shift+Enter	Insert a line break

Nested Lists

In the 2011 Formats specification, nested lists and poetry use a runover position that is two cells to the right of the right-most indent, i.e., all lines of the list (which are distinct paragraphs) must be coordinated for overall correct format, hence you must declare each nested list.

To create a nested list from an ordinary list (and from a single-level list or a pre-2011 nested list), highlight the entire list (just one list at a time), and click Nested List in the Do menu. The list becomes a Block and shows with a light blue background and all runovers are dynamically coordinated. Items in the list can be re-indented as desired (Ctrl-Tab and Ctrl-Shift-Tab are handy during input) and the runovers throughout the list will remain coordinated.

When entering a list, begin entering the first list item and then click the Style button and select N-list and level 1. Press Enter at the end of each list item to extend the list. Use Ctrl-Tab to

move the indent to the right; use Ctrl-Shift-Tab to move the indent to the left. (You know it is working when the added material is blue in background color.)

To enter non-list prose, press Enter to start a new paragraph and click the Style button and select the format desired.

Find

The Find dialog box supports "Find", "Replace", "Insert Before", "Insert After", "Ambig Trans" (Ambiguous Translation).

Find	Searches the entire document or a selection for the given phrase (one or more cells or one or more characters or for a pattern match to a regular expression). The search may be modified by "Case insensitive" (for characters) or by "Word" for cells or characters.
	Top/Cursor/Next highlights one matching phrase starting from the given start point
	Find All highlights all matching phrases
Replace	Replaces a phrase found by Find, or replaces a unit in a selection, with the given phrase. If a Find phrase is being replaced that was identified by matching a regular expression, the replacement phrase may use the subexpression notation to form the replacement phrase, otherwise the replacement phrase must be all braille cells or all print characters.
	Top/Current/Next searches for a phrase matching the Find specification starting from the given start point and then replaces it, or it identifies the next selection unit following the given starting point and replaces it in its entirety.
	Replace All searches for all matching phrases and replaces each one or it replaces all units in the current selection.
Insert Before	Insert Before operates similarly to Replace, except that the original phrase is retained in full and any new phrase is inserted at the beginning of the original phrase. When the Find phrase is located by matching a regular expression, the new phrase may be composed using sub expression notation.
Insert After	Insert After operates similarly to Replace, except that the original phrase is retained in full and any new phrase is inserted at the end of the original phrase. When the Find phrase is located by matching a regular expression, the new phrase may be composed using sub expression notation.
Ambig Trans	Ambig Trans searches for those braille elements that, although not necessarily wrong, do not back-translate to the prose they represent. This search may or may not be of value. Braille is by nature ambiguous in certain situations. Unlike V1, V2 maintains print text for the entire document while it is open (V1 regenerated print text on the fly on demand as you scrolled through the file), and then when the file is saved as a .B2K type, print that can be regenerated from braille is discarded while the print that cannot be regenerated (i.e., print for those words this search feature will report) is stored in the file. The result is persistence of the original print without enlarging the .B2K file except for the

| ambiguous elements (which are usually few in number). |
|---|

Annotation management

Most discretionary document control is handled via "annotations" in V2 very much as was the case in V1. The annotation locations continue to be displayed using small yellow triangles. Various commands (especially "Do" commands) create annotations as needed, and as in V1, the annotations usually act and get planted at the position of the cursor, sometimes taking into account the region that is highlighted.

Document control not handled via annotations relates to properties of paragraphs (i.e., "Styles"), and those are handled via the Change Format dialog box and the Style button.

V2 has some new ways you can work with annotations after they have been created, as follows.

Visibility

The Display dialog box (via Adjust and Display) is identical to the one in V1. There are two options "Show layout annotations" and "Show all annotations". If both of these are unchecked, the yellow triangles do not show.
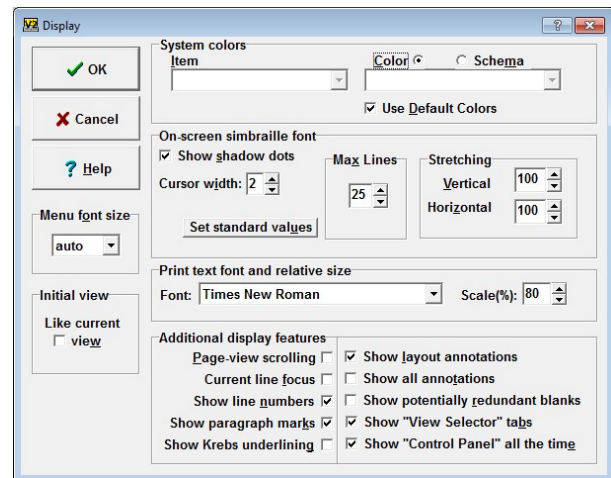
Hover

As in V1, if you hover the mouse over a yellow triangle, a message box appears, announcing the kinds of annotations at that spot.
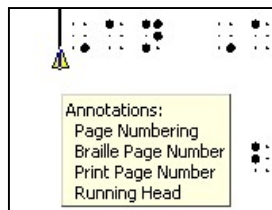
Hover (definition): move the mouse pointer to the spot and then cease all mouse movement for a few seconds (do not click any mouse buttons).

Annotation interrogation and control

With the mouse pointer on a given yellow triangle, click the right-hand mouse button (note that there is no mouse pointer shape change in V2: the arrowhead shape is used for everything). This will display the Annotation Inspector, visually tied to the given yellow triangle.

The dialog box will show one subpanel for each annotation at the given location (in the snapshot, four annotations are revealed).

Click in one of the subpanels to activate that part of the dialog box, and that will provide buttons for "Change" (to edit the properties
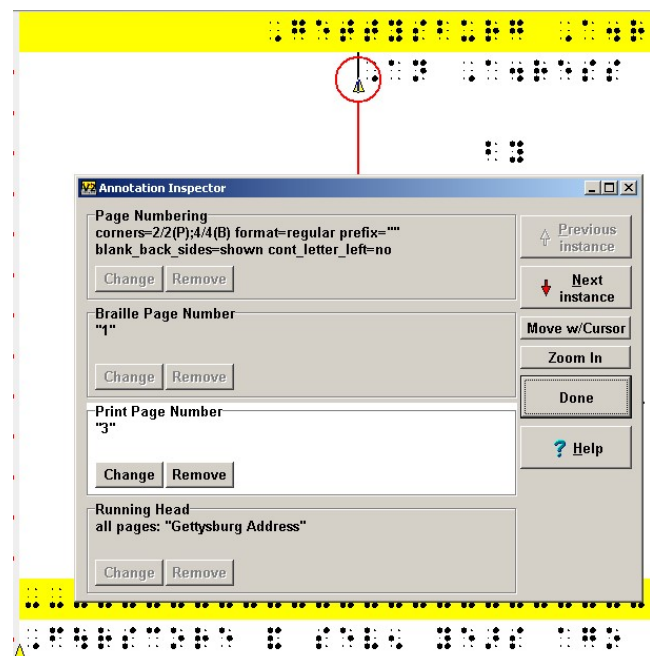
of the annotation), "Remove" (to eliminate the annotation, "Previous Instance" (to navigate to an earlier annotation of the same type), "Next Instance" (to move similarly to the next instance), and "Move w/Cursor" (reposition the annotation via arrow keys).

This snapshot shows the above dialog box after clicking in the third subpanel (but not on any button). Now the Change and Remove buttons apply only to the Print Page Number. The Next Instance button will reconnect the Inspector to the next print page number annotation (which happens to lie in this file just below the dialog box at the beginning of the last braille line shown). The Previous Instance button is grayed because there is no previous instance of print page number.

Move w/Cursor

The "Move w/Cursor" button is an on/off toggle, that when "on" activates the arrow keys for dynamically repositioning the selected annotation (Print Page Number in this case).

In this snapshot, Move w/Cursor is "on" and the down-arrow key has been pressed, moving the Print Page Number annotation down to the third braille line. Further motion via arrow keys is possible until Move w/Cursor is turned "off".

In this case, because the Print Page Number annotation controls the manner of print number expression, the motion caused a page turn indicator line to be generated (and as a result the annotation is on line four because line three is the page turn line).

The three other kinds of annotations are unchanged, still on line two. They could be moved individually if desired.

When finished, click the Done button.

If you need to change what the annotation does, you click the Change button. In this case, that brings up the Page Numbering Control, focused on the Print Number only (it is a portion of the Page Numbering dialog box).

You can also adjust page layout annotations without using its yellow triangle, via right-click on the yellow page element that was produced.

In the example shown here, right-click to the page turn indicator line shows the popup menu shown here.

The "Inspect" function displays the Annotation Inspector (you can use this method to inspect an annotation even when yellow triangles do not show

12
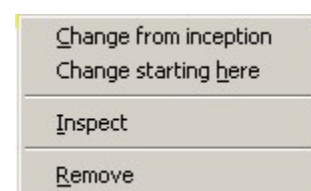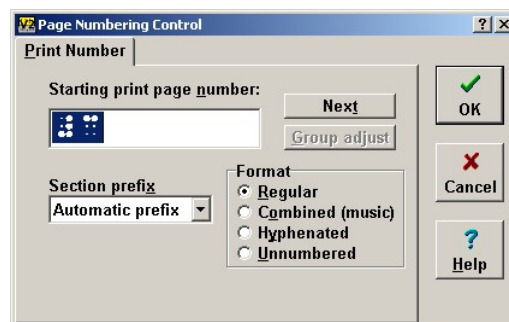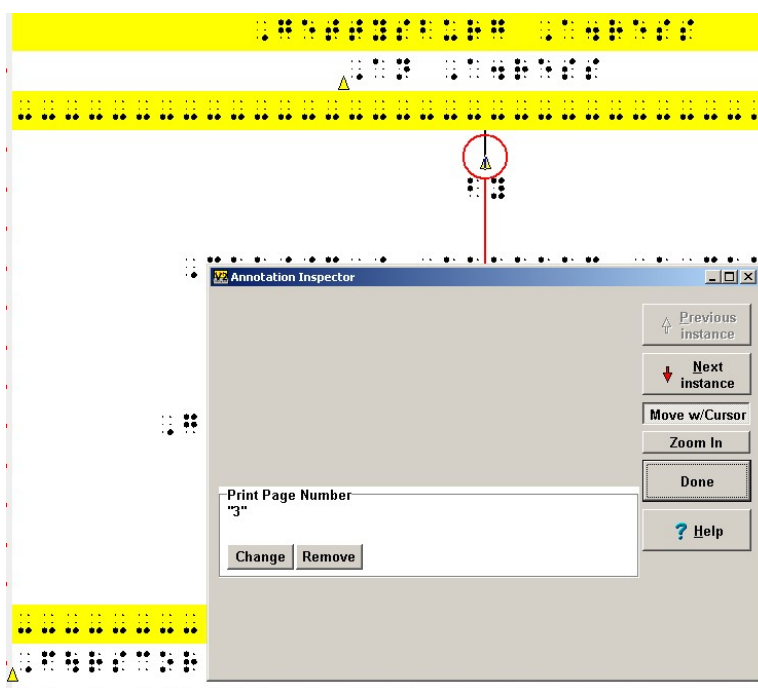
on the screen, but it works only for layout annotations, those that generate yellow braille on the page).

"Change from inception" lets you adjust the original settings (for example, correcting all running heads going all the way back to the first page on which they occurred due to the same running head annotation).

"Change starting here" will plant an overriding annotation at the current location (for example changing a running head at the current location without altering those that came before).

"Remove" will " delete the annotation, possibly affecting many earlier pages. If you want to know where the annotation is that you might remove, use Inspect to see it and then use the Remove button in the Annotation Inspector.
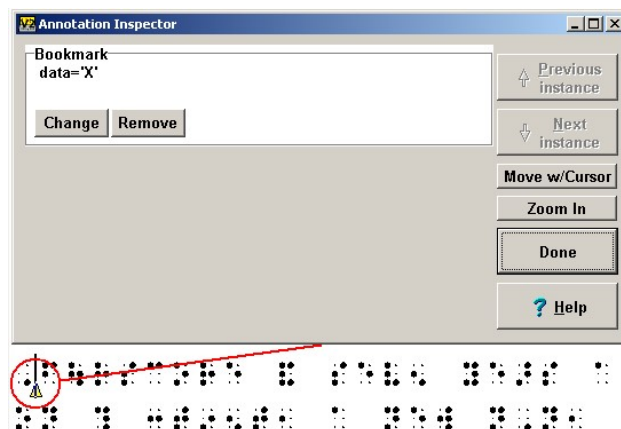
Control panel

The Control Panel is a tool bar that sits to the left of the View Selector (the sidewise-reading tabs that let you select which document to view) which is in turn to the left of the Editing Panel (optional line numbers, paragraph marks, braille or print prose).

The Control Panel can be hidden to reduce the Braille2000 window footprint. (This is done via View and Control Panel.)

New in V2 are the buttons "Tools" "Math" "rtf" "Import" "BkMark" "Insert" "Do" "A" "S" and four small emphasis buttons B I S and U.

The buttons Tools / Math / rtf select what shows in the space of the Control Panel. Because the Math and rtf panel data may be intense, in V2, the width of the Control Panel is adjustable, by dragging the separator line between it and the View Selector right or left.

If you hold down Ctrl as you click the rtf button, the rtf markup panel appears to the right of the Tools panel: you can see them both.

Note that when working with a NIMAS file, the rtf button reads "dtbook" and the panel contains extensive markup data for the NIMAS file.

"Import" replaces both "Insert" and "Append". The new "Insert" is Insert from the Edit menu.
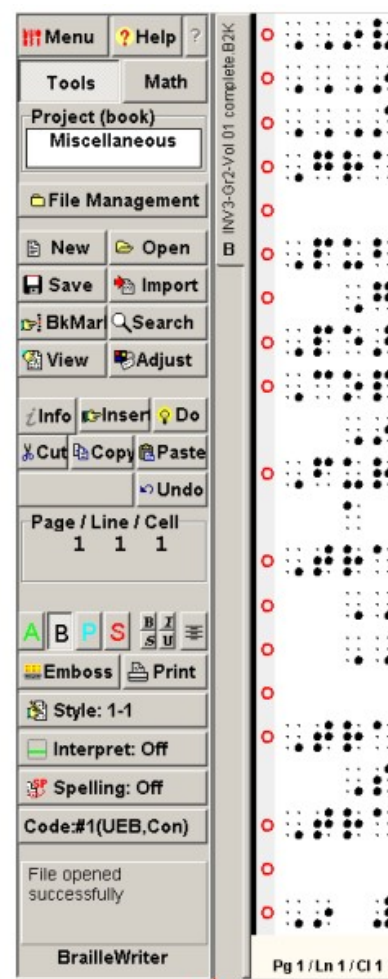
Adjustable

"Do" functions can now be performed more easily via the Do button (this should simplify how-to tutorials).

"BkMark" does X/Y/Z and general bookmarks.

There are four (smaller) toggle buttons B I S U for bold, italics, script, and underlined.

Boxing Lines

V2 has dynamically-sized boxing lines. Such lines adjust their length automatically to allow for page or line numbers and automatically resize when the document page width is changed.

To insert a boxing line, use the Insert button or use Edit / Insert /Boxing line in the main menu, and then select the type of line you want:
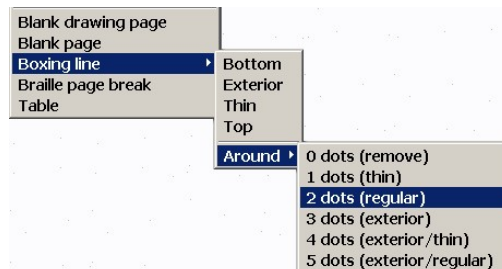
Type	What you get
Bottom	a line of (1245)'s plus blank line after
Exterior	a line of (123456)'s
Thin	a line of (25)'s
Top	a line of (2356)'s plus blank line before
Around	top and bottom lines enclosing what was highlighted

0 dots (remove)	boxing lines removed
1 dots (thin)	(25)'s before/after
2 dots (regular)	(2356) before, (1245) after
3 dots (exterior)	(123456)'s before/after
4 dots (exterior/thin)	(123456)(25) before (25)(123456) after
5 dots (exterior/regular)	(123456)(2356) before (1245)(123456) after

Note: not all patterns have a role in Formats

Tables

To insert a new table, click Insert and select Table. You get the dialog box to the right. Set the number of columns and rows and click "Create table".

The number of columns and rows is based on the print copy and the top row (column headers counts as a row). If the first row is column headers, put a checkmark in "First row is column headers" and you will get a separator line (in columnar format), otherwise there will be no separator line.

A new table of the given dimensions will be created with each field set as per the "New fields filled with" setting.
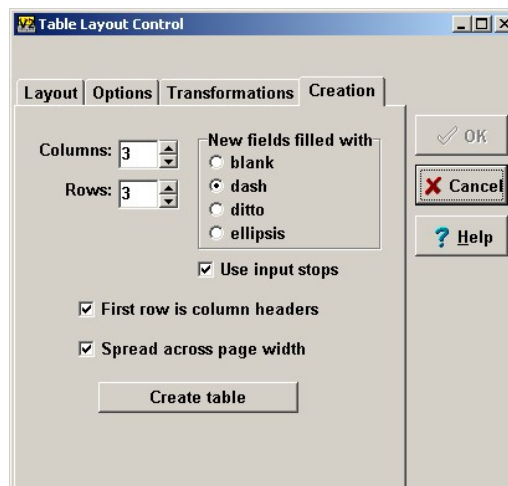
If you are *not* creating a table that will remain empty, you probably want the "Use input stops" option that will create an input stop at each field (use the Tab key to visit each input stop, one at a time, and enter data).
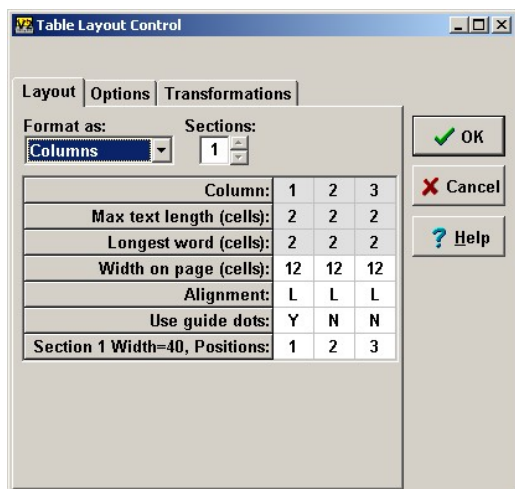

After you click "Create table" you will see the Layout panel as shown at the left. For small simple tables, just click OK.

An empty table of the dimensions you gave will be inserted into the file. The rust-colored areas (nine in this example) are the input stops, one in each empty field of the table. The tab key selects and highlights one input stop at a time, moving through all of them in a circular fashion. For data entry, you press tab, an input stop is highlighted, and you braille the value of that field. Then press tab to go to the next field until the table is all filled in. You can also fill in a table in Print View, just the same way.

The background color of a table is light blue, because it is a "block" with special behavior. Other block elements in V2 are nested lists and drawing spaces.

Note that the filler (dash/ditto/ellipsis) is code dependent.
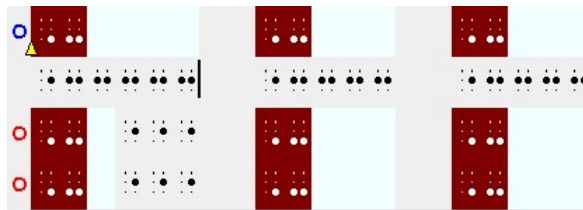
Editing Via Keyboard Entry

In the ASCII view and the Braille view, keyboard input is naturally six-key. You can change this to ASCII-braille typing via Adjust and Keyboard. In the Print view and the Source view, keyboard input is always by typing. You can enter a single cell or trigger a Speedbraille™ key by entering Ctrl-k followed by a single six-key cell or by a Speedbraille trigger (as in normal six-key mode).

Pressing the Enter key when the cursor is in the middle of an existing paragraph introduces a paragraph break at that location, splitting the paragraph into two adjacent paragraphs. Both paragraphs have the same indent and runover settings as before. If you split a paragraph in the ASCII view or the Braille view, the braille cells split at the cursor and the fragments are each back-translated into print according to the language setting. If you split a paragraph in the Print view or the Source view, the print letters split at the cursor and the fragments are each translated into braille according to the language setting. If you split a word or phrase that has print emphasis, both fragments have the same emphasis and the resulting translated braille fragments will each manifest this emphasis. Splitting a word in the ASCII view or the Braille view does not replicate composition signs and if the word split was italicized, the first fragment would continue to be italicized (it would keep the italics indicator) while the second fragment would not. Thus there are subtle differences with the behavior of the Enter key between viewing modes.

If keyboard entry is performed while a single phrase is highlighted, the highlighted material is first deleted and then the keyboard cell or letter is inserted in its place. If a multi-highlight is showing (i.e., a multi-part selection), keyboard input merely generates beep sounds and nothing is either deleted nor entered. You may delete all elements of a multi-part selection using the Delete key and after that is done keyboard input will be inserted at the location of the cursor.

Page Control (braille page break and more)

You will find Page Control in the Do menu. One function of Page Control is to manage Page Break. The choices "To next front page" and "To next back page" are available only for interpoint files (use Adjust and Document to set the embossing mode).

You can break to the next page and suppress the running head and/or the print page number and/or the braille page number. If you turn on "No print reference" the print number does not show and it does not increment automatically.
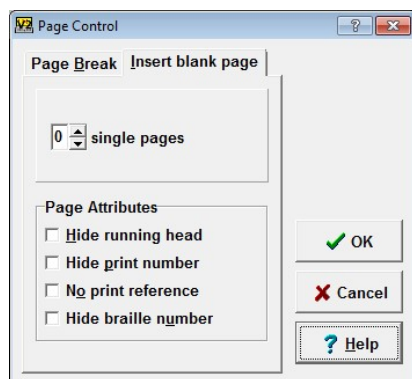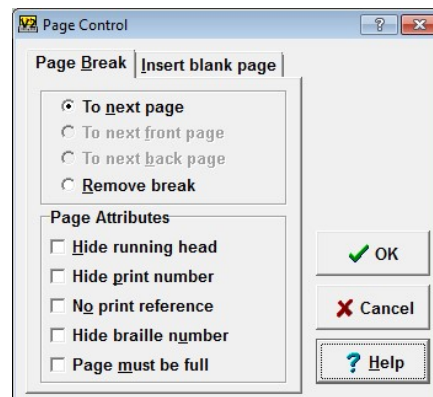
The option "Page must be full" added mid-page blank lines so that the last prose will fall on the last line (as desired on a book title page). To use this option, the following page should also be anchored by a page break.

If you click the "Insert blank page tab" you will see the second panel, shown to the left.

This function inserts a given number of blank pages (in an interpoint file, a number of blank sheets, blank on both front and back). You can specify a number from 1 to 9 and you can cause certain page layout elements to be suppressed on the blank pages.

To insert one blank page (one blank front/back pair in interpoint), you can click the new "Insert" button and select "Blank Page".

Drawing Spaces

The drawing space feature is pretty much like that feature of V1, however some aspects of it are being simplified (the feature in V1 would sometimes get confused… it was too complicated). To insert a single whole-page drawing space, click the "Insert" button and select "Blank Drawing Page".

Source Files (RTF files)

You open an RTF file via the "Source file" option (from the Open button). You then select the braille code you want to use and Continue to a Open-type dialog box. The current release supports two kinds of source files: rtf and xml (NIMAS). Source files by nature have an internal "markup"

mechanism of some kind. Borrowing from the formalism of xml, the kind of markup is identified by a "Doctype" (document type) identifier that announces the semantics of the markup mechanism. The Doctype identifier is what appears on the button at the top of the control panel, i.e., "rtf" or "dtbook".

Source file type	Filename suffix	Doctype name
Rich Text file	.rtf	rtf
NIMAS file	.xml	dtbook

Other source file types will be supported in the future.

After you choose which RTF file to process, Version 2 creates a new braille document, sets the language attribute, opens the file, translates it, and shows it on the screen. The filename suffix will be .B2K, the new type of standard output file created by Version 2.

When source files are opened, there is a document-type-specific source "panel" that is available on the left. Version 2 has several panels (toolbars) that share the same space on the left. "Tools" is the familiar assembly of handy buttons. "Math" is part of the new (not-yet-available) "math tools optional feature" (the Math button is grayed out for the moment). When you open a source document with a Doctype (document type formal specification), you get a markup panel named for the document's Doctype, namely "rtf" for RTF files; and "dtbook" for NIMAS files.

Rich Text data can also pass into Braille2000 from the Windows Clipboard.

Speedbraille™ Keys

V2 offers Speedbraille™ Keys but their nature is somewhat different than those of V1. A major change in the internal mechanism was mandated by security changes and bugs in Windows 7 & 8 & 10. In V1, Speedbraille keys (a kind of "macro") were done by recording and reusing keyboard data, literally as if a robot with lightning input speed retyped the definition whenever you triggered a Speedbraille function. It was because of this basic nature that mouse effects did not "record" (you had to do menu functions via the keyboard). That has all changed.
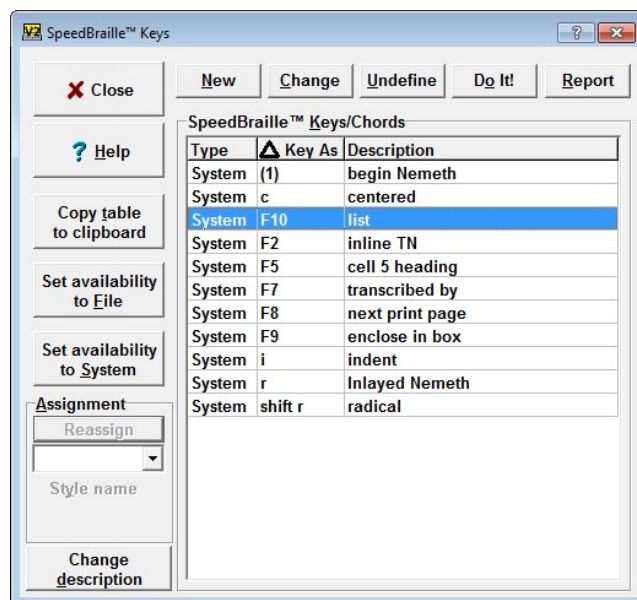
V2 does not record keyboard data as such. Instead, V2 generates internal transactions and records those. Each cell of six-key input is a transaction, as is each use of a menu function or dialog box. Because the system is generating and storing transactions, the V2 Speedbraille mechanism can "record" mouse-generated actions (such as menu use and dialog box setup). This gives a more natural behavior. But… the kinds of information recorded in the definition of a V2 Speedbraille key are totally different than in V1: there is no way to load your V1 Speedbraille definitions (sorry!). You will need to re-record them to make them work in V2.

The Speedbraille procedure looks essentially the same. **Click View and select Speedbraille Keys to get the Speedbraille Keys listing** (above). You can also click Adjust and SpeedBraille Keys.

In V2, you can reassign a definition to a different trigger key, and it is much easier to change the type from File to System and vice-a-versa. You can select one or more entries in the list (hold down Ctrl to add additional items to what is highlighted) and then click Set Availability to File or click Set Availability to System to change all entries that are highlighted.

Then click New to begin defining an new key or to change an old Key.

The sequence of panels is the same as in V1:
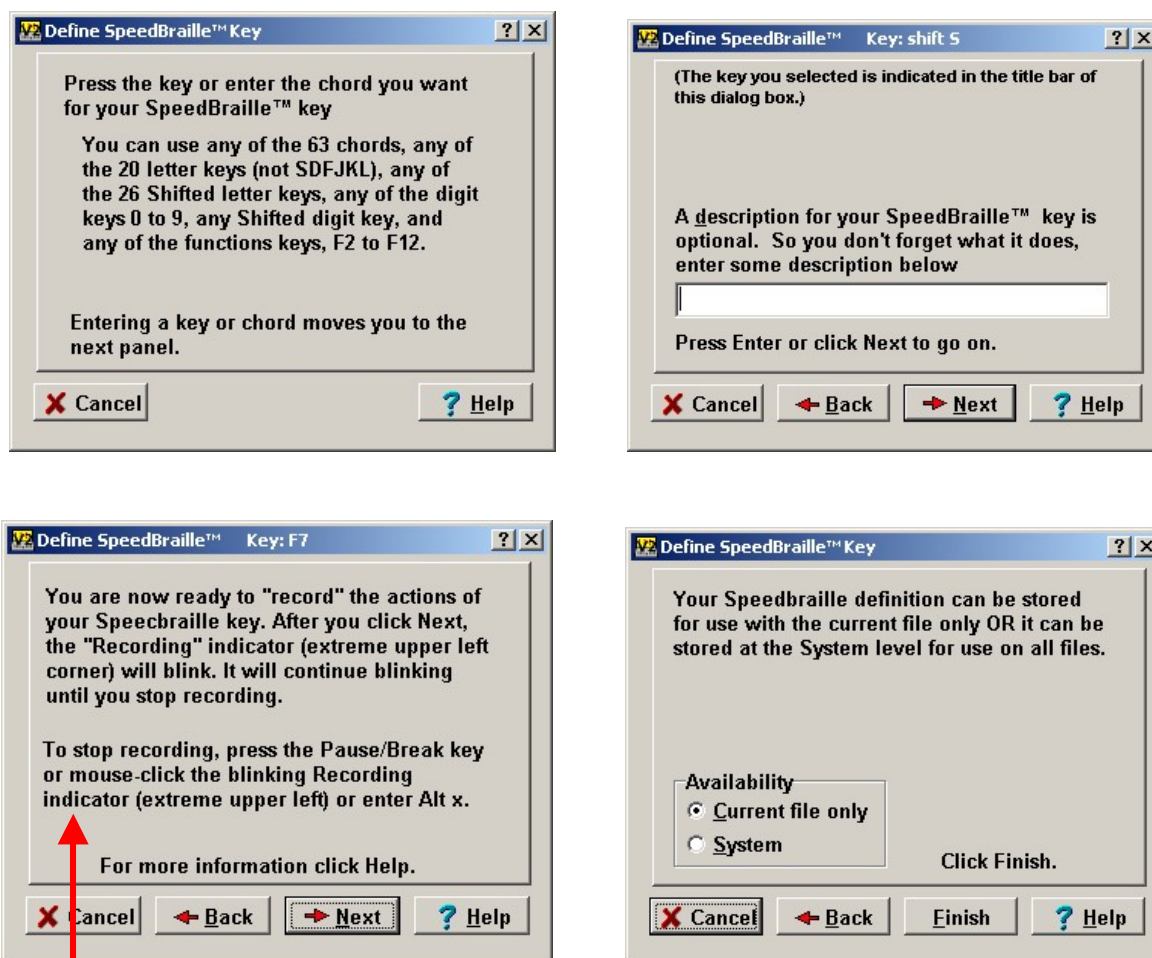
While "recording" you can now do things with either the mouse or the keyboard.

Note that you can stop recording via the mouse by clicking the "Recording' indicator (in V2 it is in the upper-left corner of the screen) or by typing Alt and x.

If you look at a "report" you will see cryptic transaction statements rather than cryptic keyboard data. V2 has a "scripting" mechanism that (in the future) will let you combine transactions (the same transaction notations that comprise Speedbraille macros) into stored routines (a better place for seldom-used but important macro behaviors). There is also an intimate relationship between V2 Speedbraille macros and V2 user-defined paragraph styles.

Copy Table to Clipboard Use this to put the Speedbraille listing on the clipboard, to paste into WORD or Wordpad to print out for reference.

Set Availability to File Click one or more entries (to highlight item) and use this button to change the storage class of all items at once.

Set Availability to System Click one or more entries (to highlight item) and use this button to change the storage class of all items at once. To import V2 Speedbraille definitions from another computer, Save a .B2K file at the originating computer, then open that file at the target computer, view the Speedbraille Keys listing, select the File type entries you want to grab, and use this button to change their storage location to "System".

Reassign Use this to change the trigger key for one selected item. Click the down-arrow to view a listing of the possible trigger keys. In that list, chords are listed by dot numbers in parenthesis (a "chord" is six-key input in which dot keys and the spacebar are pressed simultaneously).

It is not uncommon to have Speedbraille macros to effect paragraph styles. Such macros behave exactly like custom paragraph styles (defined via the Paragraph Style dialog box), and in V2 they are in fact the same thing… you will see custom paragraph styles in the Speedbraille list.

If you select such an entry (a macro that only effects a paragraph style), then in the list of trigger keys, a final entry, "StyleName" will show. This is not actually a trigger key but is the way to access the stylename property. Select "StyleName" and then a name box will show up. Enter a new name to have this function show (in the Style selector popup) as a custom style. You can use this function to convert between a named style and a key-triggered Speedbraille macro that does that style.

Change description Click to edit the description on the selected line (similar to renaming a file). Press Enter when done editing.

Interpreter line (green line)

V2 has four ways of looking at prose: ASCII, Braille, Print, and Source. Print and Source are the same except for constraints on the vertical dimension (print view maintains the vertical spacing of the braille).

Correspondingly, the green line has four display options, A, B, P, and S (in the green line P and S are identical). By right-click to the Interpret button, you get the dialog box shown here. Into "Display setting" you can put one or more display type letters (ABPS) in the following ways:
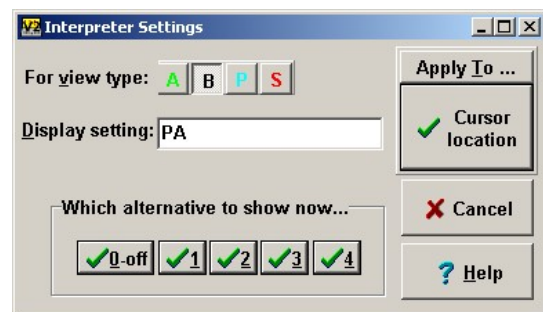
a. A single letter yields a single-line green line.

b. Two or more letters written together (such as BA) produces a green line with as many lines of prose as there are letters, one line of each kind, e.g., Braille followed by ASCII.

c. Multiple such expressions may be given separated by commas, and each element of such a setting represents one of up to four alternative behaviors. The Ctrl-g key and mouse clicks to the Interpret button will cycle though the alternative formats. For example, BA,PA has two alternates each with two lines: Braille and ASCII, or Print and ASCII.

d. There are separate settings for each of the file view types, i.e., there are separate green line behaviors for when the screen shows Braille and when it shows Print. For example, you might like the setting P for a Braille View (main screen braille, green line print) at the same time as a setting of B for a Print View (main screen print, green line braille). There is a separate setting for each of the main screen views (ASCII, Braille, Print, Source).

e. All of the above settings are stored per language, i.e., there can be different green line behaviors for English versus Nemeth.

The five checkmark buttons in the dialog box are for the immediate selection of "Off" or one of the four alternative modes. Clicking the Interpret button cycles through the modes (if the setting defines multiple modes), Ctrl-g cycles through the modes, but via the buttons 1, 2, 3, 4 you can directly switch to a particular mode. You can make a Speedbraille key to select a particular mode.

Below is a green line with the setting PA. The setting AP would put ASCII on top.
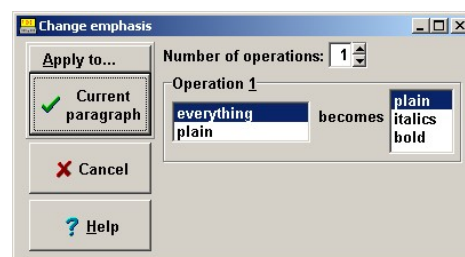
Emphasis Control

By default, source file text emphasis (italics, bold, underlining) is retained during import. There is a new way to adjust emphasis after import: click Adjust and select Emphasis.

The box to the left of "becomes" selects the existing emphasis setting. The box on the right selects what you want the emphasis to become. The box on the right can select one of plain, italics, bold. The box on the left lists all the kinds of emphasis that are in the selected application scope, which can be "paragraph", "highlighted text", "rest of file" or "all of file". The scope can be changed via the "Apply to…" button.

Emphasis changes during the importation of RTF files can be established using percent directives, described below.

More About RTF Files

You open an RTF file via the "Source file" option (from the Open button). You then select the braille code you want to use and Continue to a Open-type dialog box. (The current release supports two kinds of source files: rtf and xml (NIMAS)).

After you select the RTF file you want to process, Version 2 creates a new braille document, sets the language attribute, opens the file, translates it, and inserts the braille into the new file. The filename suffix will be .B2K, the new type of output file created by V2.

You can also Copy RTF data to the clipboard and Paste it into an existing file. In Version 2 you do not need to use "special paste" (it can usually figure out whether the clipboard contains braille or print, and it will translate print to braille; if the clipboard is such that it is unclear whether the material is braille (which comes in as-is) or is print (which is to be translated), Version 2 will ask you.

In the markup panel (i.e., dtbook or rtf) each line is a "node" in the style hierarchy. Each line is linked to the relevant location in the braille text. "p" denotes "paragraph". For example this document begins with two paragraphs in Normal style followed by one paragraph in heading_1 style. By clicking on a node name, the cursor will jump to the corresponding location in the braille panel (which is called the Editing panel) *depending on the current "cursor transfer" setting, described below*.

Using the Markup Panel for Summarization, Navigation, and Control

Summarization

Via the Options button and the Reveal selection, you can add to the nodes in the Rtf panel a variety of useful information, such as the print text or the print page number (even both at the same time if you wish). Remember that RTF files do not of themselves support print numbers (you would need to augment such a file with %pn= directives and perhaps take advantage of page breaks).

You can change the width of the toolbar by dragging its right-hand border left or right. If you use Reveal to show auxiliary information, you may need more room. A new wide-screen monitor or notebook helps give extra width.

Navigation

When you click on a node in the Rtf panel, the cursor jumps to the corresponding spot in the Editing panel. If you hold down the Ctrl key as you click, the corresponding segment of the document becomes highlighted.

You can also do the reverse: click in the Editing panel and have the Rtf panel indicate (via bold linking line) the corresponding node in the Rtf hierarchy. These behaviors are controlled by the "cursor transfer" buttons in the upper right corner of the Rtf panel.   There is one button for right-to-left transfer (Editing panel position to Rtf position) and another for left-to-right transfer. The buttons toggle.

Control

You can use Ctrl and clicking in the Rtf panel to select multiple nodes (representing multiple regions of the file, as you click to select in the Rtf panel, corresponding text areas become highlighted in the Editing panel). You can also select nodes based on node name and/or node depth.
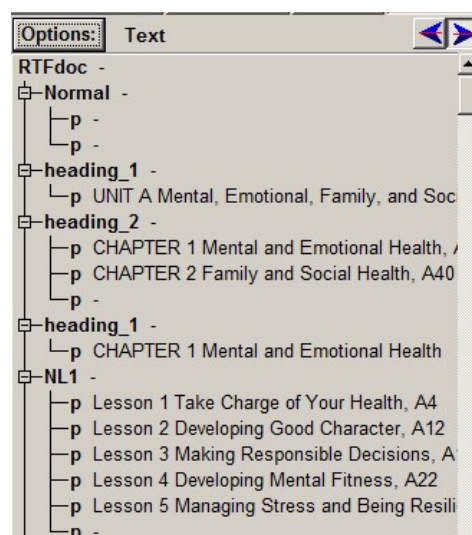
Having established a multi-selection, you can do things to the selected text, such as change style, set guide word text, format contents entries, etc. This notion of "select and act", using multi-selection capability, is unique (in braille processing) to Braille2000.

You can form a multi-selection in the Editing panel using the mouse to highlight text, while holding down the Ctrl key.

Import control via %Directives

"%Directives" (also called Translation Typing Directives) are transcriber-added cues for the Braille2000 import mechanism for RTF files. They act as a sort of additional markup. Each directive begins with the percent sign, and thus they are called "%directives". Their use is optional. Note that print text coming from the Windows clipboard is also RTF and thus these directives can also be applied via the clipboard.

There is only a vague relationship between the page layout of a typical print document and the standard layout of braille pages. In most cases, at some point human judgment is needed to establish the desired braille page arrangement. Such layout changes can be done (a) to the braille after importation into Braille2000, or (b) to the print file, in ways (unique to Braille2000) that will import with good braille page layouts.

There are six text to braille layout relationships which can be used (besides also using percent directives). **These relationships guide the braille format when doing implicit format imports.**

1. Centering versus left-justified
   Text which is centered generates centered braille. Text which is Left-justified generates normal (left-justified, non-centered) braille. Other text alignments, such as right-justified should not be used.

2. Italics versus bold versus underlined versus normal font
   Emphasis in print is transferred to the imported braille.

3. Indent and Paragraph (left) Margin
   The amount of paragraph indent and "runover" (in text, one uses the term left margin, but it is the same concept as runover) determines the indent and runover of the resulting braille. The text indent and left-margin settings should be set in jumps of quarter inches, with each eighth-inch corresponding to one braille cell position. The table below shows the indent and left-margin settings for popular braille paragraph forms. Note that the left-margin measurement is taken from the left side of the printable area on a page and thus does not include the page margins.

Braille form	Text indent	Text margin
1-1	0"	0"
1-3	0"	0.25"
1-5	0"	0.5"
3-1	0.25"	0"
3-3	0.25"	0.25"
3-5	0.25"	0.5"
5-3	0.5"	0.25"
5-7	0.5"	0.75"

This equivalence table is for the default scale of 8 cells (of indent/runover) per inch of print margin.

The code %cpi$v$ can be used to change the scale, for example %cpi4 would double the inch values in this table.

4. Spacing
   A paragraph in an RTF document can have space above or below the prose. This is the "Spacing" setting given (usually) in points for space above and below the text. When this spacing is large, then if the paragraph is centered or structured like a cell-5 or cell-7 heading, the use of spacing cause it to import as a heading style with the necessary blank lines before and/or after. A "large" amount of space before or after is an amount at least ¾ of the text size in points. If the text is 12pt size, then the paragraph spacing must be at least 9pt to be noticed in this way. To manifest a braille centered heading, there must be enough space before and after and the text must be centered. To manifest a cell-5 or cell-7 heading, there must be enough space before, and the first-line indent must be zero and the left margin must be ½" (for cell-5 heading) or ¾" (for cell-7 heading). The spacing amounts are paragraph format parameters in Word or Wordpad, blank lines before or after do not count. You can define a Word style to create such paragraph effects, if you wish.

5. Tabs
   Tab stops work much the same way, with each 1/8" of text space mapping to one cell position.

6. Choice of font
   You may use any font you like, with the following two provisions:
   a) Elements of text that should be set in Computer Braille Code can be set in Courier font;
   b) Elements of text that are literal ASCII-braille symbols can be set in a simbraille font, such as "Braille from BRL2000".
   All other fonts are translated to the target language, set in advance of translation via the Braille2000 "Language" button. See also the %cbc= and %brl= directives.

7. Keep together
   Paragraphs with "Keep Together" or "Keep With Next" settings in print retain those behaviors when imported as braille paragraphs.

You can do an entire file using implicit formatting (without %directives affecting indent and runover), as described above. As an alternative, you can adorn your document with one or more %directives, listed below, to help manage the process of translating it into formatted braille. A %directive is placed alone on a line or at the beginning of a paragraph (it is followed by one space if not on a line by itself). It specifies an overriding braille format for the immediately following text and all subsequent text that has the same text layout (same print-implied indent and left-margin, and same style (centered versus left-justified)). When the text layout changes, implicit formatting resumes, unless (a) you use another format directive, or (b) you have used %explicit to switch to "everything-is-explicit" mode. In "explicit" mode, braille layout is determined *exclusively* by directives.

When explicit control is used throughout the document, text paragraphs might as well all be blocked at the left margin (0" indent and left-margin).

There are thus three ways to prepare a print document:
1. Implicit Layout (the default behavior)
   by carefully laying out the print text (perhaps by applying WORD styles), making sure the indent and left-margin are right for braille,

2. Hybrid Layout (using %directives just where needed)
   by laying out the text in a good-looking way, and using a format directive whenever additional control is needed, and
3. Explicit Layout (put %Explicit at the top of the document)
   by leaving prose layout as-is and indicating all braille layouts via %directives.

In the hybrid approach, a directive that affects indent or runover applies to the following print paragraph and all subsequent print paragraphs that have the same *print* layout. When the print paragraph layout changes, the effect of the directive terminates (but if using %explicit, the effect endures until the next layout directive that changes indent and/or runover).

Directives you can add to the RTF file ("percent directives")

There are a few aspects of page layout and translation that are unique to braille. These are indicated by special directives which can appear inline (in the text) or as a separate paragraph. A special directive is a single word that begins with the % sign. When written inline (with the text) it is typed as a word, set off (as needed) from any adjacent text by blanks. The following special directives are defined. Italics denotes a variable (fill-in) part of the directive.

Beginning in version 2.276, you may invoke your own names style via the directive %stylename. See the discussion below for *%stylename*.

Directive	Use
%explicit	To ignore paragraph shape and rely only on directives for braille format control
%implicit	To honor paragraph shape (indent and margins) and to format the corresponding braille paragraph according to the scale of 1/8" per cell (you may use directives below as required but some effects wear off)
%block	To set the following paragraphs as 1-1 braille paragraphs
%block*v*	To set the following paragraphs as block paragraphs of level v
%bls	To turn on blank line separators, added whenever the indent of a new paragraph matches the runover of the previous paragraph. Use %tb to turn this off.
%bn=*number*	To give the braille page number in textbook format (lower right) The number may have a letter prefix of t or p to denote transcriber or preliminary pages, respectively. If you use this directive do not use %ln= described below.
%box=*letters*	To insert a boxing line. Letters are TMB (top/middle/bottom) optionally with E (exterior), such as %box=T or %box=ET (note ET and TE act the same)
%brl=*keyword*	To take print as literal ASCII-braille when *keyword* is part of the fontname (default setting is brl=Braille)
%cbc=*keyword*	To translate print into computer braille code when *keyword* is part of the fontname (default setting is cbc=Courier)
%center	To cause the single following paragraph to become centered braille
%centered	To cause all following paragraphs to become centered braille

%cpi*v*	To set the equivalent number of cells (for indent/runover) per inch of print margin
	The default value is 8.
%em*p=b*	To map print emphasis (*p*) to braille emphasis (*b*)
	p and *b* stand for one or more of the letters ibu (italics bold underlined) used singly or in combination without spaces. The letter p (plain) may be used alone for braille emphasis. When fewer than three letters are used, the emphasis not mentioned is implied: for example i used alone implies also ib iu ibu (i.e., all forms of italics) whereas iu implies also iub (all forms if underlined italics). In some cases, you may need to use multiple %em directives to get the mapping you want, and then the order of such directives may be important: for example %emu=i %embu=p together in this order would cause all print underlining except bold underlining to be braille italics. Unless otherwise mapped, other patterns of emphasis are the same in braille, for example, in this case non-underlined bold would be bold, etc.
%footer	To cause the single following paragraph to be used as the running footer
%footer=off	To stop showing a running footer
%footer=*v*	To activate an automated guide-word footer, *v* can be 0 or 1 or 5
	If *v* is 1 or 5, the guide words are the first word in cell-1 or cell-5 paragraphs on the page, respectively. If *v* is 0, all guide words are indicated via %gw (see below).
%g*v*	To change the language (braille grade) to number v, where v is
	0: literal ASCII-braille (source text is not translated at all)
	1: uncontracted form of the current language code (if such is possible)
	2: contracted form of the current language code (if such is possible)
	3: computer braille code (CBC) only in BANA context
	%g without a digit returns to the previous code (same as %L)
	Note: for a document in a foreign language (always uncontracted), 1 and 2 do nothing
%gw*v*	To indicate what word (or words) are to be guide words
	Place %gw (set off by a space before and after) immediately in front of the word in a glossary entry that is to be used as the guide word. If more than one word is to be used, give the number of following words, as in %gw3. If you have used %footer=0 then you must use %gw to tag all guide words that you want to be active. If you have used %footer=1 or %footer=5 then you can optionally use %gw to indicate alternative and/or additional guide words that are active.
%head	To cause the single following paragraph to be centered with a blank line before and after
%head*v*	To cause the single following paragraph to be a heading
	Possible values for v are 1,2,3,5,7 and they generate centered, cell-5, cell-7, cell-5, and cell-7 headings, respectively. (When used as %head5 or %head7 the operation is identical to %subhead5 and %subhead7, respectively.)
%hide	To cause all following text to be ignored (hidden from the transcription)
%hide=off	To terminate %hide, subsequent text will not be ignored
%indent	To set the following paragraphs as 3-1 braille paragraphs
%indent*v*	To set the following paragraphs as indented paragraphs of level v
%interpoint	To set the embossing mode to interpoint (double-sided)
	When used, this directive should be at the beginning of the document.

%L	To set the language back to its previous value (if any)
%L*keywords*	To set the language via keywords
	The same behavior as two items down, but without defining any language number.
%L*v*	To set the language to language number v
	Note: the given number must have been defined previously (see below), or else you will be prompted to select the definition during translation. v must be a positive number (%L0 and %L (no keywords) are equivalent and will reinstate the previous language).
%L*v=keywords*	To set the language (for automated translation)
	Keywords separated by commas and without any spaces are used to set the language for translation and define that choice as language number v so that this same language can be selected throughout the remainder of the document by number only.
%left	To cause all following paragraphs to be regular (uncentered)
%list	To set the following paragraphs as 1-3 braille paragraphs
%list*v*	To set the following paragraphs as list items of level v, i.e., %list1 for the first level, %list2 for the second level. This forms a multi-level list (as per 2011 formats) until you use a non-list directive (such as %indent).
%ln=*number*	To give the braille page number in literary format (upper right)
	The number may have a letter prefix of t or p to denote transcriber or preliminary pages, respectively. Note that %bn= and %pn= described elsewhere in this section are for textbook format–do not mix the two: if you use %ln= in your document do not use %bn= and %pn= and vice-a-versa.
%math	To take the following paragraph as MathML (XML) markup
	This feature is pending.
%note	To set the following paragraphs as 7-5 braille paragraphs
%np	To assert "next print page" in textbook format
	For this to act you need to have used %pn= previously to assert print page numbers. %np is then used before the first word of a new print page to increment the print page number (and generate a print page turn line as necessary).
%off	To turn off all directive processing in the remainder of the file
	This directive should be placed on a line by itself. After this setting, % is translated as an ordinary character. And thus there is no way to reestablish directive processing (because all subsequent directives are treated as merely plain text). For the occasional leading percent sign (percent following a space as in %data) add a leading % to handle it as a normal character, as in %%data (see %%text below).
%page	To generate a braille page break
	Special effects for the new braille page are given by added letters and numbers. %page= is used to define the effect of a hard page break in the print document. (see below)
%page*v*	To generate a braille page break to a front or back page position (works only in an interpoint document (see %interpoint)
	The digit *v* can be 1 or 2 or 3. %page3 is the same as %page. %page1 breaks to the next front page (inserting a blank page as necessary). %page2 breaks to the next back page (inserting a blank page as necessary).

%page*h*	To generate a braille page break and apply certain optional attributes to the new braille page
	Optional attributes can hide certain page elements on the new page and the page can be set as a "filled page" (see below). The letter r hides the running head. The letter p hides the print number. The letter b hides the braille number. (Number hiding does not change the page number sequence.) The letter s suppresses the print number (the new page does not participate in the number sequence, as if it doesn't exist). The letter f turns on page filling (the final prose will always fall on the last line of the page (blank lines are added as necessary to make that happen). When necessary you can follow %page with more than one attribute letter and you can also add a digit (1-3) as mentioned above. For example %page1rp would break to the next front page position and hide both the running head and the print number.
%page=npp	To have each hard page break in print imply %np
	Like %np (described above), print page numbering needs to be established using %pn=. The document may use both hard page breaks and %np directives to advance the print page number, as appropriate.
%page=bpb	To have each hard page break in print imply %page
	You should normally use %page=npp or %page=bpb but not both, however one does not cancel the other: if you use both then each hard page break implies both %page and %np until you use %page=ignore.
%page=ignore	To do nothing at each hard page break in print
%pbl	To start preserving blank lines in print (as blank lines in braille)
%pbl=off	To stop preserving blank lines in print (the default)
%pn=*number*	To give the print page number for textbook format (upper right)
	Note: if also using %bn=p1 (e.g., to start preliminary pages), put that directive before %pn=. To have print page numbers in your document you need at least a starting %pn= directive. After that, you need to signal each print page start, either via hard page breaks and %page=npp, or by inserting either %pn= or %np just before the first word of each print page. For unusual print page numbers you will need to use %pn= in a manner mentioned just below.
	Special Print Numbers
	The print number may be in the form prefix-number, as in 10-17 or Intro-1. In UEB (strange as it is) a numerical prefix is transcribed as a preceding number with its own number sign (and no distinguishing punctuation, i.e., it will read 1017 in print or #AJ#AG in braille, unfortunate). But at other times, 10-17 might denote spanned pages and need the hyphen in braille.
	When a numerical section number is used (no hyphen in braille), replace the hyphen with the vertical bar (|) as in %pn=10|17, this will generate the proper braille number #AJ#AG (no hyphen).
	When a hyphen is desired (spanned pages) use the hyphen as in %pn=10-17. Additional page number behaviors may be indicated by following the page number with a colon and a special symbol, as follows:
	Colon 1: for section-number hyphen single-page-number: %pn=Ref-8:1
	Colon 2: for spanned pages transcribed as a whole: %pn=3-4:2

Colon Letter: to give the continuation letter (rarely needed): %pn=6:c
If the page number is non-numerical or mixed, enter it as given, such as %pn=One or %pn=Intro1. (In some cases the "Next Print Page" function will be unable to increment such a number.)

	See Also Print Number Extras (elsewhere in this document).
%pn=off	To cancel the generation of print page numbers
%reqsp	To check for a blank line in braille at this point. Same as %reqsp1.
%reqsp*v*	To check for a blank line in braille at this point, generating one if necessary %reqsp is like %space except where the blank line would be the first or last on a page or where a blank line already exists by other means.
%runhead	To cause the single following paragraph to be taken as the running head Note: be mindful that a running head must be short in order to fit on but one line of braille.
%runhead=off	To cancel the use of a running head
%space	To generate a blank line in braille
%space*v*	To generate v number of blank lines in braille
%*stylename*	To set the following paragraphs as per *stylename* (a named style, i.e., a name shown in the named styles list in the Paragraph Style dialog box). The stylename may identify a built-in style, a user-modified built-in style, or a user-defined style. If the stylename matches one of the %directives listed here, the standard definition applies (if you create your own named style, use a name that differs from all %directives defined here). For example, you can create a named style called "space" but you cannot assert it via %space because it conflicts with a standard %directive (you could name your style "sp" and then you could use %sp to use it).
%subhead	To cause the single following paragraph to be a 5-5 paragraph with preceding blank line
%subhead*v*	To cause the single following paragraph to be a v-v paragraph with preceding blank line Use %subhead5 to generate a cell-5 subheading. Use %subhead7 to generate a cell-7 subheading. (Other values for v also work but are not standard heading choices.)
%symbolslist	To declare the following items to be Special Symbols List template data See discussion at Automatic Special Symbols List (elsewhere in this document).
%symbolslist=off	To terminate the section of symbols list template items
%tb	To start "tight blocked paragraphs" i.e., to turn off %bls (blank line separators between certain paragraphs). Beginning in version 2.269, this option is on by default.
%tnl	To generate a beginning transcriber's note indicator in braille
%tnote	To set the following paragraph as 7-5 braille transcriber note with beginning and ending indicators
%tnr	To generate an ending transcriber's note indicator in braille
%toc	To set the following paragraphs as contents entries
%toc*v*	To set the following paragraphs as contents entries of level v (v=0,1,2,3, etc.)
%toc=off	To end special handling for contents entries
%*x-y*	To set the indent (x) and runover (y) properties of the following paragraphs For example, to achieve a standard 3-1 paragraph style, use %3-1.

%%text   To use the % character as plain text, use %%% to represent % (the only concern is leading percent signs, as in %something in the prose)

Importing the RTF file

In Braille2000 V2, you can use Open / Source File to import an entire RTF document and create a braille version of it. You can still use Insert or Append to import by adding the translated text to an existing braille file. You do not have to create a new empty braille file first: you can simply Open the RTF file as a Source File.

You can also paste text from the clipboard by clicking the Paste button. Via right-click to the Paste button, you can import RTF material using customized options (see Special Paste, below).

Using the %L directive

The %L directive is needed only in RTF material that is composed of multiple languages, such as a mix of English and French, or where portions are contracted and others are uncontracted (for switching between contracted and uncontracted translation, the %g directive may be used). The directive may be used in defining "special paste" to set the code. Note that "L" stands for "language" although in Version 2 the term presently used is "code".

The %L directive (written here with capital L for clarity, but like other directives it is not case sensitive, i.e., you may use %l just as well) can be used in two main ways:
1. To declare the language for major sections of the document, as in
      %Lenglish,contracted
      block of text in English
      %Lspanish
      block of text in Spanish
      %Lenglish,contracted
      continuing in English

2. To declare a phrase within other prose that should be translated in a foreign language (note that in many contexts isolated foreign words are transcribed with dot-4 notation (i.e., as English) and do not use the foreign language codes), consider
      Then he said, %Lfrench Sacré Bleu %L and hung up.
      In the above line, the %L by itself returns to the prior language (presumably English).

You identify your choice of language by writing %Lkeyword,keyword,keyword… using as many keywords as needed to make your choice clear. No spaces are permitted in the entire %L directive. If you write %Lenglish, you may sometimes get unanticipated results depending on context because Braille2000 has several languages whose name contains the keyword English. %Lenglish,iceb,contracted or %Lenglish,iceb,uncontracted are better.

There are some built-in biases for selecting the code to use: (a) if both contracted and uncontracted options are available (and you don't indicate which in your %L directive), the contracted code will be used. If your document uses UEB and you use %L to switch to a foreign language and you do not indicate which code group to use, the ICEB group will be used (the same group as for the previously selected code). Similarly, if your document uses EBAE and you use %L to switch to a foreign language with specifying a group, the BANA group will be used.

29

As Braille2000 operates, you may see code keyword abbreviations in the Code button, such as "con" for "contracted". Those abbreviations may also be used in the %L directive. You will also see that the codes used in an existing file are numbered (#1, #2, etc.). When importing more RTF material into that file, you may use %L#*n* to re-select the nth code already in use in the target file.

Using the Code dialog box (right-click the Code button to see it), you can name your own codes (so-called "Quick codes") as synonyms for a system code. For example, if you define "EBAE" as your own code name, you can use %L"EBAE" to select it from within the RTF document (assuming that you have already defined that name). Quick code names (your assigned names) are always written within double-quotes, as shown.

Special Paste Manager

Use this panel to define menu items for Special Paste

Menu caption: Blocked   Remove

Data type: Rich Text (RTF)

Parameters: %explicit %1-1 %tb

✓ Define   ✗ Cancel   ? Help
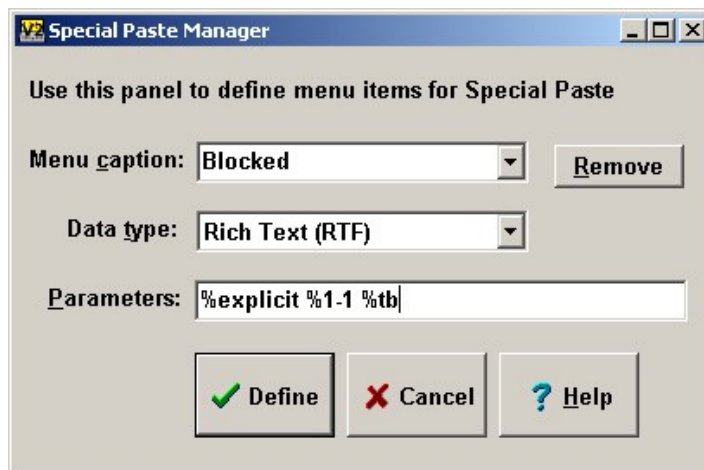
When a document contains a great quantity of language changes, you may define a language number to minimize the directive. Language numbers run from 1 up (and need not be used in sequence, i.e., you can use 23 without having used 1 through 22). A language number must first be defined via the directive %Lnumber=keywords, as in %L3=English,Uncontracted. This directive sets the designated language and defines it as language 3 which may then be used simply as %L3. Once the definition is made, you can use %L3 wherever you would use the longer directive, for example in %L3 a phrase that is to be uncontracted %L and then normal material.

If you use %Lnumber without ever defining what that language number is to be, you will be prompted to select a language to define that language number, while the translation is in progress. Once defined, %Lnumber (for the same number) will indicate the given language throughout the document, during translation. Once translated, language numbers are discarded from the braille file, but language transitions, affecting back translation and spelling checking, endure, and can be spotted as "Lang" annotations.

When preprocessing a print document with comprehensive tools, such as MS Word, you may be able to construct time-saving macros to insert %L directives in an automated or semi-automated manner.

Special Paste

If you right-click the Paste button, you get the "special paste" menu of choices. If you select "Define RTF Paste" you will get the dialog box shown here by which you can name additional special paste menu choices and give an initial line of percent directives to pre-condition the behavior of the import process.

For example, you could create the choice "Blocked" (as shown to the right) that makes simple blocked paragraphs suitable to prepare the label for the spine of a bound volume. The codes %explicit %1-1 %tb are given, that (in the absence of contrary codes in clipboard data) would form simple 1-1 paragraphs without separating blank lines.

You could also have special paste choices preset for foreign language material by giving the appropriate %L directive. If you give multiple directives, leave one space between them (just as you would do if you put the directives into the RTF file itself). The parameters are handled as if they constituted an implied first line of the RTF file. You may define as many special paste behaviors as you wish.

NIMAS Files

The dtbook button shows the NIMAS markup panel. (dtbook is the DOCTYPE for all NIMAS files). Had you opened an RTF file, the button would read "rtf" and would show the "rtf" markup panel. The Tools button shows the "Tools" panel, the conventional Braille2000 set of control buttons. You can adjust the relative widths of the panels by moving the gray divider bar left or right with the mouse (see the small (red) upward-pointing arrow above).
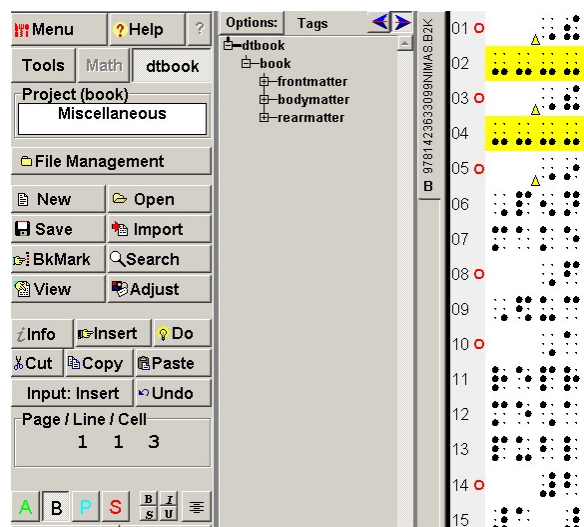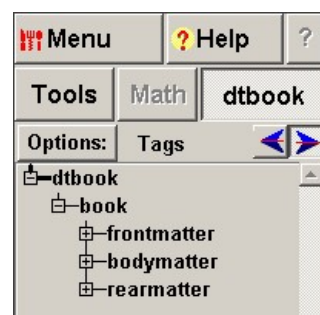
If you click "dtbook" (or "rtf") while holding down the Ctrl key, you get a two-panel display as shown at the left. The "Tools" panel is on the left and the "dtbook" panel is to its right. Of course when you have two panels on the screen, the editing area is reduced in width.

There are four ways to view the NIMAS document, using the four buttons shown at the bottom of the snapshot at the right: A (ASCII-braille); B (braille); P (print); and S (source). ASCII, braille, and print always show braille page geometry. Print view items that would distort the braille page geometry are shown as simbraille cells, for things like tall fractions or tall math expressions, and source images. The source view shows document images (if they are available via link) and all math expressions, but because of size differences, the source view may not show all lines on a braille page (you can use vertical scroll to see what does not show initially).

Entering Ctrl-w toggles between the last two view-as settings. The manner of viewing can be adjusted by the View button's View As choices or by the four view selection buttons shown at the bottom of the Tools panel (image above left). Note that Ctrl-w can toggle view-as even when the Tools panel is not showing.

Navigating using NIMAS tags

The dtbook panel first shows like this (see above right). Most of the panel is used to show the tags hierarchy, initially just "dtbook" and "book". Click on ⊞ to reveal item detail. Click on ⊟ to hide detail. When you click on a tag name (not on the + or -), the editing panel will scroll to the corresponding location in the text, depending on the cursor-transfer buttons.

The cursor-transfer buttons show either "up" (inactive) or "down" (active). A click changes the buttons state.

Clicking in the dtbook panel sets the focus node (a bold line traces the path from the focus node to dtbook). When the right cursor-transfer button is down (active): the click will also scroll the editing panel to the corresponding place. Clicking in the Editing panel moves the cursor. When

the left button is down (active), the click will also scroll the dtbook panel and open nodes as necessary to set the focus on the corresponding node. You can set either or both cursor-transfer buttons to the up (inactive) position when you want the Editing panel cursor and the dtbook panel focus to operate independently, for example, so that the dtbook panel does not scroll while you are doing editing via the Editing panel.

If you click on a tag name in the hierarchy while holding down the Ctrl key, you toggle highlighting of that element, in both panels.

If you drag the mouse, vertically, across multiple elements in the hierarchy, you can highlight elements and their corresponding text. The screenshot below was taken after clicking some ⊞ points to open some elements and then two elements were selected, either by clicking with Ctrl or by dragging the mouse across them. The result is highlighting in both panels (see screen-shot below).
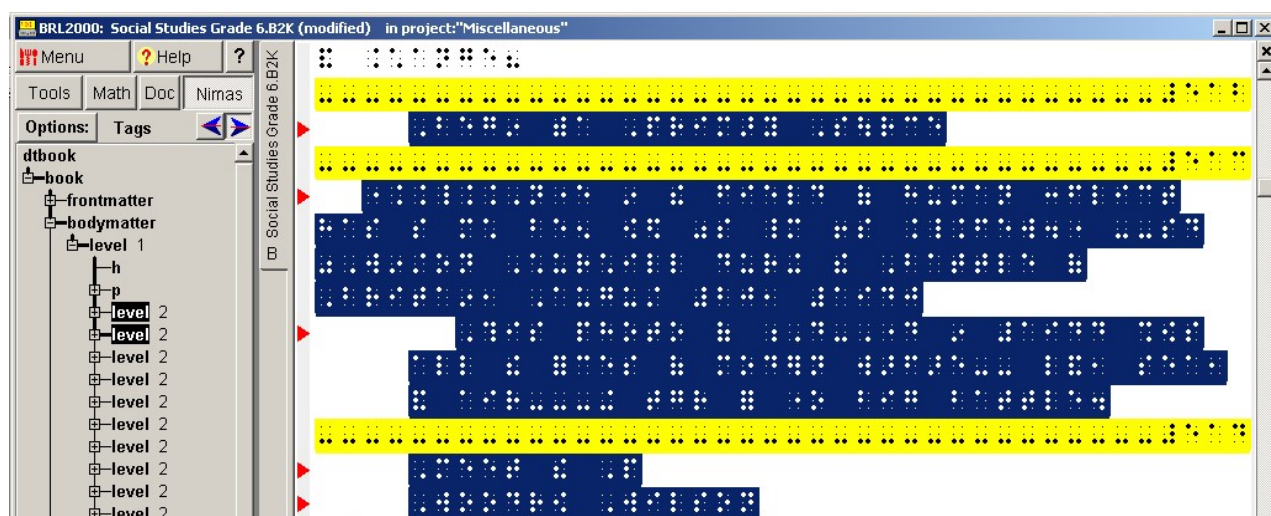
Browse the print copy
With a new textbook project, you may want to flip through the print copy while noticing how the publisher has arranged the NIMAS elements. You can switch to the Print View and work your way through the file, on screen, as you flip through the print copy of the book. This is a good way to get a feel for tag semantics. If you can't figure out the way a tag is used, you can consult the brief tag name definitions at the end of this document. Many of the tag names are reasonably mnemonic. And besides, you don't care what they mean, only how they seem to be used, relative to format changes you plan to make.

Reveal auxiliary element information

The "Options" button at the top of the dtbook panel presents a popup menu with various panel control settings. One of those is "Reveal". The initial setting is "Tags only". But you can add a variety of additional information. For example, select "Text" and you get the display at the left.
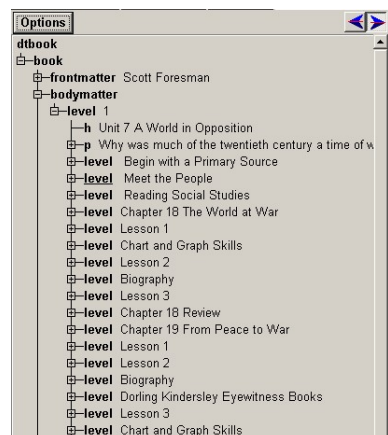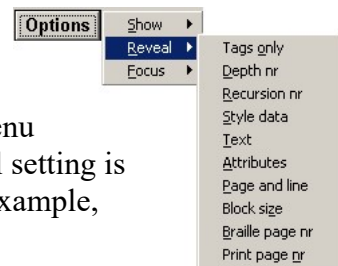
Now you can see the leading prose for each element, as shown at the left.

Tag names (also called element names) are always shown in bold. Additional data, as selected by the Reveal options, are shown in regular-weight text.

The table below lists the information you can display in the Dtbook panel, and what you might do with it.

"Reveal" settings (multiple choices may be active simultaneously)

Tags Only	Just the element tag names show
Depth nr	How deep (how far to the right) the node is in the hierarchy relative to the leftmost "level" node. Depth number is useful for selecting headings of a particular level.
Recursion nr	How many times the same kind of element occurs within itself, e.g., a list within a list. An element with the same tag name does not have to be immediately contained in another element—there may be intermediate elements of some other name, such as a list item "li" that is contained in a list but has another list within it—the list element is still said to have recurred, even when the sequence is list, li, list.
	Elements that do not recur receive a recursion number of zero (you could argue that they in fact recur once, but using zero gives an inspired conceptual fit to braille format concepts for multi-level things like exercises and lists). Elements that recur two more times are numbered, by tag name, from the top of the NIMAS hierarchy outward. The recurring element nearest dtroot is numbered 1; the other recurrences receive accordingly higher numbers.
	A publisher may describe exercise material as a "list". If the list element has a recursion number of zero, it is like a single-level exercise. If the list element has a recursion number of 1, 2, or 3, it is like a multiple level list, with the appropriate indent.
Style data	The braille style name or indent-runover characterization of the paragraph at the given node.
Text	The print text at the given node. Useful when relating Dtbook panel entries to the print copy of the book.
Attributes	The publisher's element attributes. Xml allows each element to have optional

	attributes. They are occasionally useful.
Page and line	The location where the associated text starts in the file, given as pagenumber.linenumber. The page number is the page ordinal number (as shows in the cursor position readout in the Tools panel).
Block size	The size, in cells, lines, or pages, of the text in the given element in the file.
Braille page nr	The braille page number where element text starts. Useful for navigation and when deciding where to break the book into volumes.
Print page nr	The print page number where element text starts. Useful for navigation and when deciding where to break the book into volumes, and to relate the node to the print copy of the book.

The "NIMAS Touch"™

The "NIMAS Touch" is a trademark of Computer Application Specialties Company, and refers to the special way Braille2000 uses NIMAS tag information (i.e., the dtbook Panel) for summarization, navigation, and control.

Summarization

Using the Reveal and/or Focus options, you can learn about the textbook as a whole, to help plan overall formatting decisions. For example, if you use Options and Focus and Focus By Name, you get a selection box that lists all the element names.
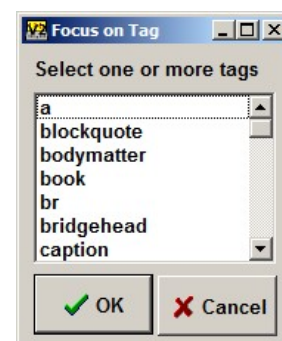
You could select all of the headings (H1, H2, …). That would let you see just the headings in the dtbook panel (the other elements still exist but are hidden; some structure elements such as "level" also show so that the heading elements have places to exist in the hierarchy). You can use Options and Reveal and Text to see the headings themselves. In this way you can get an overall understanding of the use of headings.

Navigation

By clicking in the dtbook panel you can set the cursor to the corresponding spot in the prose, and vice-a-versa, under the control of the "cursor transfer" buttons. You can use the Show and Focus settings to configure the dtbook panel and then by clicking there you can jump to that part of the prose in the editing panel and vice-a-versa.
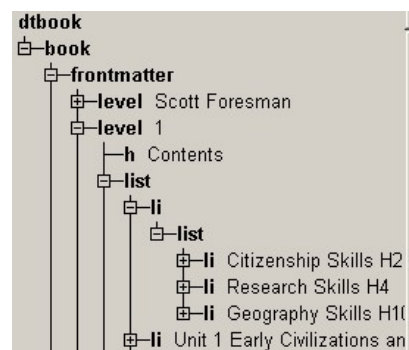
Control

By using the dtbook panel to form various multi-highlight selections, you can than carry out global operations on selected prose elements (such as change format, change emphasis, handling contents entries, guide words, diacritics, etc.).

For example, you could right-click on one H4 node and click "Select similar" and "Same tagname". This would select all H4 nodes in the entire document along with all of the associated prose for those elements. You could then right-click and use "Change style" and then select a new style: that style is applied to all selected locations throughout the document (all at once). You could also use the Style button or the style dialog box (via right-click to the Style button). The general theme is "select and act" and it is the most compelling when the select process

selects multiple locations in some automated way such as "select similar, same tagname" as used here.

Processing Contents
One area where significant variations are possible is in the markup for the table of contents. A publisher may put the page numbers in an element by themselves (e.g., "linenum") or the numbers may be plain text in a paragraph ("p" element) or a list ("li" element). Our sample textbook describes the contents as a list (actually a multi-level list, see Formatting Lists, below) in which the page number is not set off in any way, other than being at the end of each entry.

To format the contents, you need to find the contents (use the Dtbook panel), handle the entries to get page numbers and guide dots arranged properly, and then to manage the overall format (e.g., as a multi-level list). In this section we deal with parts one and two.

Display the Dtbook panel. You know contents will be an element in "frontmatter". Use Options, Reveal, Text to show the text, and then look for "Contents". You will find it is the second "level 1" element in "frontmatter". Some publishers may identify contents with the xml optional attribute **class="contents"** (one of the rare occasions when the element's attribute may be interesting). The following formatting steps do not require relabeling any elements but you can, if you want, change the tag to, say, "contents".

Ctrl-click the node that represents the entire contents. This may be a "level" node or you may have changed its tag to "contents" via the above (it makes no difference). Do not click the "h" node, that is merely the heading line. Now the entire contents is highlighted. Click Menu, Do, Line Number to bring up the Line Number Control dialog box. Click the tab "Auto-Contents". For the contents arrangement in our sample textbook, the default settings are good. Click OK. The contents-entry line number mechanism is then applied to everything highlighted and all of the page numbers become contents line numbers, with guide dots.
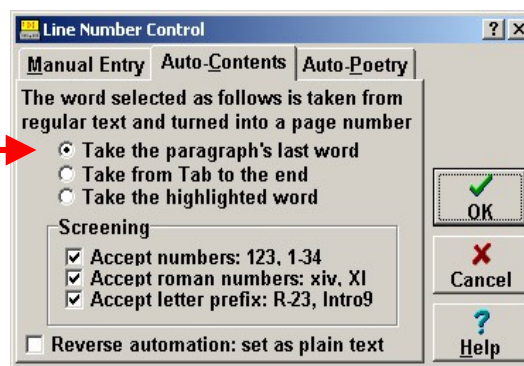
Set guide word text
A good NIMAS file will use the tag name "dt" to denote each guide word or phrase. You can select all of those via the Dtbook panel and then use "Set guide word text" (in the Do menu) to mark all selected items as tagged guide words.

When the "dt" element comprises a long phrase, it obviously won't fit in the guide-text footer (particularly when there are two entries in the footer). In the dialog box, the "I want to edit phrases longer than xx." can be used to "catch" those guide word entries that are too long. For each one you will be shown a dialog box in which you can arrange the abbreviated guide text for that entry. This can be very tedious, but automation

36
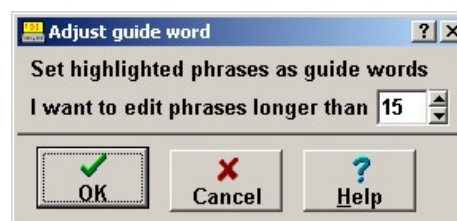
of this task would not likely do a good job   (If you don't shorten the entry, Braille2000 will merely truncate the entry to fit the space of the footer line… not very nice.)

Transferring work to V1

While cute and hopefully compelling, V2 is young and it may therefore be temperamental at times. Yet work must get done. There are two techniques you can use to push past any impertinent software behavior:

Edit As Plain Text

For tables, tabbed prose, and multilevel lists, you can highlight such a block of material and click Menu, Edit, and "As Plain Text". Doing this will leave the braille unchanged but will remove all tabs and fancy automation, leaving you with plain text that you can then adjust in any way you wish. Note that some layouts (especially columned tables) will lose column-wrap and line-wrap, becoming a sequence of one-line paragraphs. If you use this tool, you should do all automated adjusting first because after you apply Edit As Plain Text the pizzazz will be gone.

Save As Annotated Braille Text (ABT)

You can save your material using SaveAs (Menu, File, SaveAs) and then set the file type to "Annotated Braille Text (.ABT)". This will give you a file you can then work on using Braille2000 V1. But to tone down the new features of V2 (things that V1 can't understand) the entire file will be converted to "plain text" as per the above paragraph. The plain text conversion affects tables, tabbed prose, and multilevel lists. The braille remains unchanged but for those kinds of structures, the pizzazz will be gone when you work on the file using V1.

Note that it is <u>not</u> necessary to convert your files to .ABT format just to send them out to be embossed: Braille2000 V1 is able to open and emboss .B2K files (V1 can't edit them). The .B2K file produced by V2 has the same basic properties as the .ABT file, in that it can be embossed without Braille2000 software when necessary. Production centers now accepting .ABT files for embossing should be able to handle .B2K files using normal procedures.

Select Pages

The ability to create a file copy (to save or perhaps just to emboss) that consists of certain selected pages is very useful. For example you can select just a page or two to emboss if there is a problem with the embossing of just some pages. There are also ways to select just pages containing reserved space for a tactile drawing.

With the advent of tables and nested lists (and later, footnotes and drawings), i.e., those special paragraph ensembles known as "blocks," there are new technical challenges for select pages. For example, a table in column format may span several pages. As a whole it has normal table properties, but it you select just a page that is but a part of the table, it no longer has its regular table properties—it is something else when not whole.

The new Select Pages function uses two strategies to handle blocks (i.e., tables, drawing, nested lists, and other ensembles with gestalt behavior). The first strategy is to automatically expand the page list (i.e., the selected pages) to include all of any included block structure. By this strategy if you select a page that contains part of a multi-page table, the other pages are included so that the entire table is selected.

The second strategy is to automatically apply "Edit As Plain Text" (discussed above) to convert a block that is only partially included to plain text form (the braille is unchanged but any special properties such as table attributes are discarded). As plain text, the part that is on a selected page can be reproduced independently from the rest.

If you select pages that would operate on only a part of a block structure you will see the warning message (the line in red) *unless* you put a checkmark at "Select whole block, e.g., table".

As shown at the left, there are some tables that will become plain text, in the resulting set of pages. If the task is to emboss the pages, this is always OK. If you need to do further editing the loss of block properties (e.g., table properties) will be a handicap.
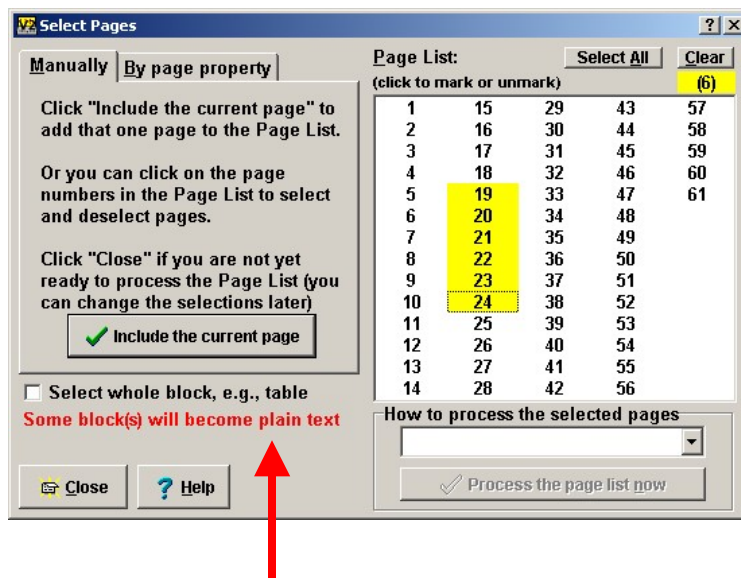
If you checkmark "Select whole blocks" additional pages will show (in yellow) in the list of selected pages, so that whole blocks (i.e., whole tables) will be selected, with all their properties intact.

You can record a Speedbraille™ Key to perform Select Pages. There are multiple strategies for doing that. Note that while a file is Open, the Page List of selected pages persists from one use of Select Pages to the next.

To Do This…	Record This…
Add the current page to the Page List	Click "Include the current page"; Click Close
Emboss the current page	Click Clear; Click "Include the current page"; Select "Emboss pages" at "How to…"; Click "Process the page list now"
Emboss page one	Click Clear; Click 1 in the Page List; Select "Emboss pages" at "How to…"; Click "Process the page list now"
Ink print page one	Click Clear; Click 1 in the Page List; Select "Print pages" at "How to…"; Click "Process the page list now"
Copy drawing pages to new workarea	Click Clear; Click "By page property"; Put a checkmark in "Page with drawing space" (only, no other checkmarks); Click "Select by page property"; Select "Copy pages to new workarea" at "How to…"; Click "Process the page list now"
Count the number of non-blank pages (the count shown below the Clear button)	(while viewing a file with at least one blank page) Click Clear; Click "By page property"; Put a checkmark in "Text-free pages" (only, no other checkmarks); Click "Select by page property"; stop recording with the dialog box visisble; then Click Close

Unified English Braille (UEB)

Braille2000 V1 predates UEB, V1 does not know a thing about that code. You will get into trouble trying to do a UEB transcription in V1. Braille2000 V2 supports both EBAE and UEB, and it can be used to convert one code to or from the other (but note that EBAE code is itself ambiguous and that converting EBAE to UEB needs human proofreading and perhaps editing; converting UEB to EBAE works well).

The V1 "Language" button is called "Code" in V2.
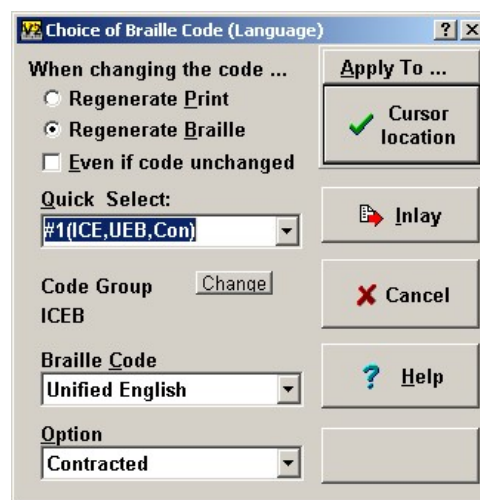
When you create a new file, it automatically uses the system default code which is UEB (you can set the default to another code if you wish). Your file makes use of one or more codes. To assign a different code to a passage, highlight the passage, click the Code button, and select the code from the list. The popup list contains just those codes already being used somewhere in your file. If you want to assert some other code not yet active in the file, right-click the Code button and use the dialog box. Note that you can royally mess up your file by doing improper code changes. For this reason, the system always has you confirm a code change. The code for a given passage designates the bi-directional equivalence between print and braille: **when you change the designated code one or the other will change… you have to choose which**. The system will ask you or you specify by the buttons at the top of the code dialog box (shown above). Select "Regerate Print" when the interpretation (green line and/or print view) is wrong. Select "Regerate Braille" when the braille itself is wrong (but the print is right).

Using the dialog box

The dialog box shown above appears when you right-click the Code button or you left-click it and choose Other. Codes are presented grouped by the controlling agency. The codes relating to UEB are in the ICEB group and this group is usually pre-selected (based on the system default code). The older (pre 2012) codes are in the BANA group. They are the codes used by Braille2000 V1. You normally do not want to mix codes from different groups… to change the group setting you must click the Change button.

Braille2000 codes are all for use in North America (there are no foreign codes as found in other countries). For foreign language study in North America, there are specific codes for Spanish, French, German, Portuguese, Italian, Russian that are based on cell usage described in the publication *World Braille Usage*. These codes are all uncontracted and use cells that would be contrractions for accented letters and/or special symbols. Prior to 2012, foreign language transcriptions were always done this way in a manner that maintaiined the punctuation and typeform indicators of EBAE. In today's world, the ICEB organization has not addressed the issue of foreign braille codes, but has given UEB a set of three-cell accented letter designations, the idea being that foreign language will be done as if it is English with "funny accents". For those who want to contiinue using special codes following *Word Braille Usage*, Braille2000 offers, in the ICEB group group Spanish, French, German, Portuguese, Italian, Russian that are uncontracted and use the punctuation and typeforms of UEB (analogous to old practices). Those codes are completely unofficial (but very useful). Note that the foreign language codes exist in

both the BANA group (where they are EBAE compatible) and in the ICEB group (where they are UEB compatible) and none of them produce braille as used in the home contries.

Translating UEB to EBAE and vice-a-versa

By changing the code and selecting Regenerate Braille you can UEB into EBAE. To do that, do the following:

1. Open the file of UEB braille. (We assume it is all UEB. If not work on each UEB section separately.)

2. If the Code button does not show Unified English, right-click that button (to get the window above) and set the code to UEB and click "Regenerate Print". Click "Apply To" and select "All Of File". (To convert to a different code, the print translation must be correct!).

3. Right-click the Code button (to get the window above) and select English (you need to go to the BANA group via the Change button) and click "Regenerate Braille". Click "Apply To" and select "All Of File".

You can use the same method to change EBAE to UEB, but remember that EBAE does not back-translate flawlessly (the print may be wrong, e.g., saying "but." for "b." in a letter-enumerated list), and thus the resulting UEB may be inaccurate in spots (meaning you should proofread it and expect to make some manual adjustments).

For additional capabilities, see the section Working With Multi-code Documents.

UEB Emphasis

UEB has four kinds of emphasis: italic, bold, underlined, and script. (script applies to decorative fonts used for special math letter symbols, it is conceptually different from the others, and not well thought out, in my opinion). Note that UEB does not support double-underlined as mentioned in the 2011 Formats manual (there is a provision for transcriber-defined special emphasis, but it consumes extra braille cells).

The concepts of italics, bold, underlined, script, capital, and grade-1 all exist in UEB as applied to single print symbols, to words, and to phrases, and each indicator in different and when being terminated early each concept has its own special two-cell termination indicator.

Furthermore, you can use symbol bold or symbol capital (or any concept) on a contraction and it affects only the first print character. Yes, you can make just the "i" in "ing" bold and still use the "ing" contraction. Contractions cease to be used only when in indicator has to appear in the middle of the contraction, such as in "ch*ild*".

The normal approach is to use a symbol indicator for a single print letter, to use word indicators for whole words up to two in number, and to use the phrase indicator with closing termination indicator for three or more words. Unlike in EBAE the terminator goes right where the effect stops, however you **can** terminate a phrase effect with a word effect or even a symbol effect, but those uses are atypical. You can use the word indicator mid-word and its effect continues to the end of the word. You can use the word indicator in a word and the termination indicator to stop

the effect before the end of the word. You can be very creative. I believe it is bad form to use a phrase indicator mid-word.

As in EBAE, the normal usage for multiple adjacent paragraphs with the same emphasis is to re-introduce the emphasis at the start of each paragraph with the termination indicator only at the end of the final paragraph. In theory the phrase indicator at the start does carry through multiple paragraphs by itself, but it is considered cumbersome to the reader to track large blocks of text from the same indicator, so it is repeated for convenience. The exception is for displayed material.

A technique that is unique to UEB is embedding displayed material between a line containing only beginning phrase indicator(s) and a final line containing only termination indicators, as illustrated figuratively below. In this approach (which normally does not span multiple braille pages), the beginning indicators are not repeated on each paragraph. (2011 Formats would also want boxing lines, most of the time, for displayed material.)

Displayed Material Example (from the ICEB UEB manual, the concept being a grid of capitalized single letters)

code for phrase caps and phrase G1

    A B C D E F

    G H I J K L

    M N O P Q R

terminate G1 code and terminate caps code
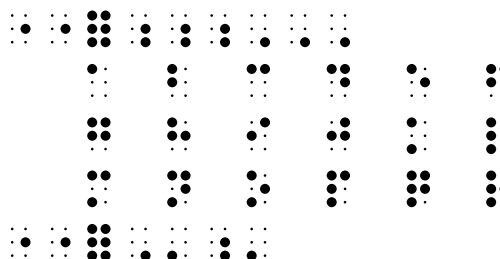
State and emphasis indicators

	Cap	G1	Number	Italic	Bold	Underline	Script*
Symbol							
Word							
Phrase							
Terminate							

*Script denotes the fancy-font letters used in math (such as $\Re$ for the set of rational numbers).

Handling a spreadsheet

Here is a spreadsheet as presented by Excel. Highlight all or a part (here just the first four columns were selected). Copy this to the Clipboard (ctrl-c) and then Paste (ctrl-v) into Braille2000.
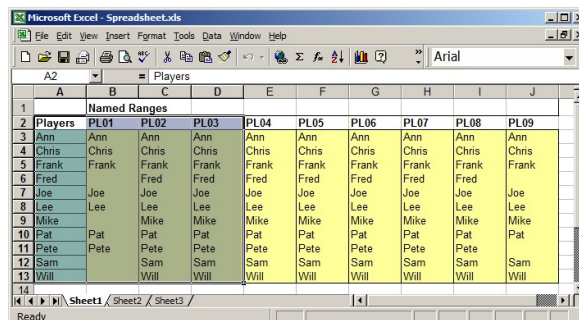
The generated braille is shown in the snapshot below.

If the whole spreadsheet were selected, the table would have appeared in stairstep format (which you could change to listed or linear, or you could arrange for vertical sectioning).

(The underlying simplicity here is that Excel will put a spreadsheet or a portion onto the clipboard as an RTF table that the Braille2000 paste mechanism already knows how to handle!)
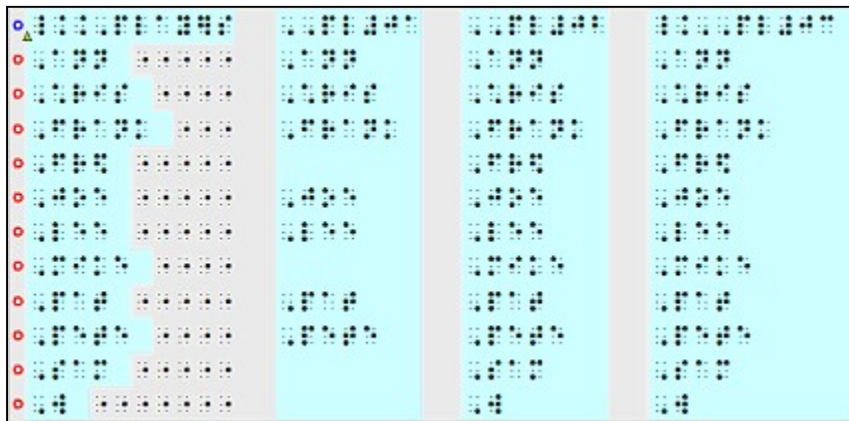
Math Toolbar Input

Note: when math toolbar input is more mature, it will be packaged as an optional extra-cost feature for Braille2000 V2. At the present time, it is free.

When the cursor is in a part of the file where the language has math capabilities (i.e., Nemeth, NUBS, or UEB), the Math panel is available to assist with math and special symbol input. (The width of panels is adjustable, so the precise layout may differ a bit from what is shown here.)

The large buttons select categories of math input items, including symbols, Greek letters, and forms, such as square root and fraction. Not all items are defined by all braille codes, and the implementation of math translation is new to Braille2000 and not all of the Nemeth and UEB math capability is available at this time. More and more capability is coming.
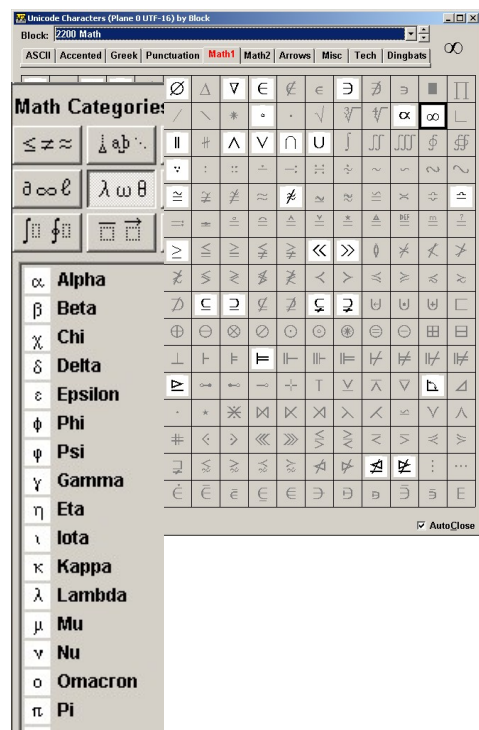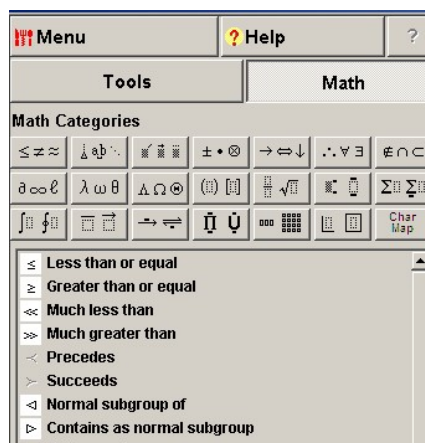
The symbols and forms that are available are shown in white (above, and left). Those in gray are not available (but of course you can use six-key input to do anything).

The char map button (above) displays a comprehensive grid of thousands of symbols (arranged by Unicode sequence, grouped into categories, left) that you can use as input for your document. Symbols available in the selected math code are shown in white and those not available (by translation) are gray. As you can see there are quite a few highly specialized math symbols that the translator does not know or the language (UEB in this case) does not define. The more customary symbols are there.

42

In the Char Map, the tabs select panels with different kinds of symbols and foreign language characters. The choice box at the top (called "Block" identifies the subcategory of the Unicode character set (much of which has no built-in braille translation, but it is still a useful resource for what some unusual symbol is, and its Unicode number (via the readout in the lower left).

When you click on a white-outlined symbol, it is inserted into your braille file at the cursor.

The more common symbols and Greek letters are handy in the main Math panel, for example, lower-case Greek letters: click the category button to see the list of choices and then click an item (a white item) to have it be inserted at the cursor.

As with filling in tables, after you click a math form, such as a vertical fraction, you get (in print and in braille) that form with placeholder entries (a and b in this case) that are input stops. When you press Tab the "a" becomes highlighted and you can type over it or input a subordinate form (say square root or a variable with an exponent). When you press Tab again, the "b" is highlighted and you complete it in the same way.
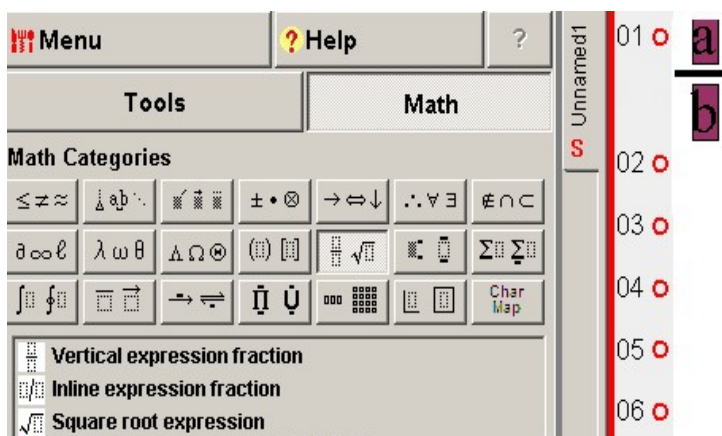
Composition is like building a table…

Suppose you want the square root of x/2. Move the cursor to where the formula should go. Click the Math button and then the [square root button] button and then click [Square root expression]. You will get the square root ox X where X is an input stop (like a blank table field). Press TAB to highlight the X portion and then click [Vertical expression fraction]. You will then have the square root of a/b with input stops for a and for b. Press TAB enter x (it replaces a). Press TAB, enter 2 (it replaces b) and you are left with the formula you want.

Step 1: the outer form:
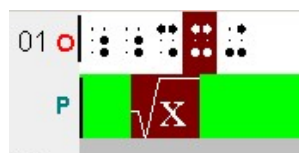
Finally, all input stops have been replaced:
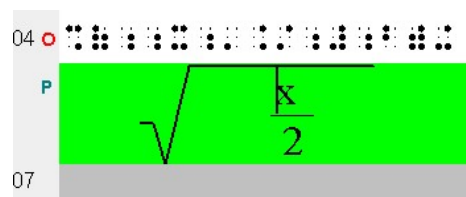
Provisional Guidance

It is because of BANA that UEB has replaced EBAE as the operational braille code in the United States. Note that BANA has no true "authority" and merely suggests practices. By "Provisional Guidance" (see their guidelines on their web site) BANA describes a bizarre approach for a math transcription by which narrative text is in UEB code while individual formulas and math expressions are in *uncontracted Nemeth* (never before heard of) with each such expression surrounded by three-cell begin/end "code switch" indicators. The result is considered by everyone as being hard to read and confusing. Nobody outside BANA itself knows what "Provisional" refers to, but it is widely known that (a) BANA operatives in Canada have eliminated Nemeth code altogether and (b) the Library of Congress National Library Service has requested bids on comprehensive training materials on UEB math code. Both of these moves suggest that "Provisional" means "until current Nemeth readers are exhausted" (perhaps six years) and then Nemeth code will be eliminated. Please note that Nemeth code is considered essential to the study of STEM field subjects, principally advanced mathematics. It is widely understood that UEB math code is not now and never will be suitable for such endeavors (it will discourage STEM field studies in blind students).

For Braille2000 to handle "Provisional Guidance", the document must switch back and forth from UEB to UN (what do we call non-traditional Nemeth in uncontracted form? we will use UN in this writeup, the setting in Braille2000 is "Provisional Guidance") many times per page, and perhaps even more than once on the same line(!). The concept for the transcription is that the main text is in one code (UEB) while other code (UN) peppers the material, like a silver inlay (that is unfortunately tarnished; Braille2000 finds this approach to braille literacy backwards, mean spirited, and contrary to common sense).
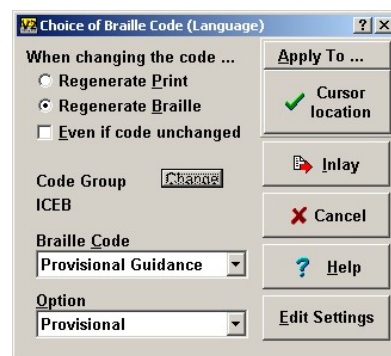
In the Code dialog box, there is a new button called "Inlay". Rather than changing the Code to the given setting, the Inlay button also includes changing back to the original code at the end of the segment. It is used in two ways: (1) you can highlight a segment of material and then use Inlay to set that segment in a new Code (this use is similar to clicking "Highlighted Text"), and (2) with nothing highlighted when you click Inlay a text token (the name of the code) is inserted at the cursor, highlighted, and then the code is changed for that highlighted region. This latter function (for which you can make a Speedbraille macro) is what you will need to transcribe according to Provisional Guidance.
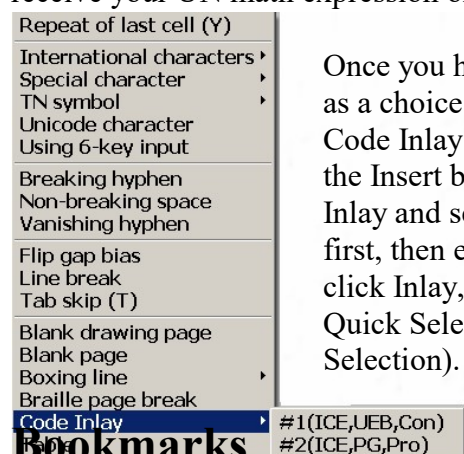
The ICEB group (of codes compatible with UEB) has a setting called "Provisional Guidance" which is the UN code mentioned above (uncontracted Nemeth) plus dynamic generation of the begin and end indicators. When you need to write a math expression (assuming you are doing Provisional Guidance), you put the cursor where the expression is to go and you select Provisional Guidance and click Inlay (or press your Speedbraille trigger key). You will then see something like this.
For this example, the word "Consider " was entered and the cursor was just after that when Inlay was done. Notice the begin/end indicators and the Code change annotations. And because

"nemeth" is highlighted, that filler will be replaced by what you enter next, i.e., it is ready to receive your UN math expression or to receive generated math from the math toolbar.

Repeat of last cell (Y)	
International characters ▶	
Special character ▶	
TN symbol ▶	
Unicode character	
Using 6-key input	

Once you have used a secondary Code in your document, it will be listed as a choice when you left-click the Code button and when you use the Code Inlay menu item within the Insert group. To use Code Inlay, click the Insert button (you then see the menu at the left), then go to Code Inlay and select the Code you need. If the code you need isn't listed at first, then either (a) right-click the Code button, select your Code, and click Inlay, or (b) right-click the Code button, select your Code, enter a Quick Select Name and then click Add Quick Name (see Quick Code Selection). After either of these, the menu will show the Code choices.

Breaking hyphen
Non-breaking space
Vanishing hyphen

Flip gap bias
Line break
Tab skip (T)

Blank drawing page
Blank page
Boxing line ▶
Braille page break
Code Inlay ▶ #1(ICE,UEB,Con)
                  #2(ICE,PG,Pro)

Bookmarks

Version 1 had three bookmarks: X, Y, and Z. Version 2 has unlimited bookmarks that you can name as letters/numbers or words or phrases. And Version 2 bookmarks can be grouped into categories (if you wish). The BkMark button responds to various mouse actions. If you left-click the button, you can jump to any bookmark in the file. If multiple categories are in use, you jump by category and then by bookmark name. If you hold down the Ctrl key as you left-click, you can both Jump To or Set a known bookmark. Known bookmarks include those used in the document plus "standard" bookmarks you declare per category (see below). Initially the known bookmarks are X, Y, and Z, and you can set them via left-click while holding down the Ctrl key. Finally, a right-click displays the Bookmark dialog box, shown here.
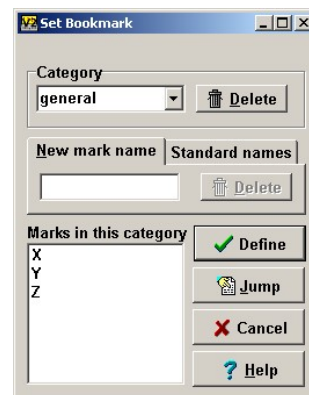
To create a bookmark, type its name into the "New mark name" box and click Define. If the bookmark is known, it will appear in the Marks list. To change its location, select the bookmark in the list and then click Define.

You can create new category names: just enter the category name in the Category box and then enter a bookmark name and click Define. "general" is the default category name. You can remove all bookmarks in a category using the Delete button in the Category box.

If you click the "Standard names" tab, you can type in a list of bookmark names using a comma to separate adjacent names. Click the "Store" button (not show until you click the tab) and those bookmarks will become "known", i.e., they will show up when you hold down the Ctrl key and left-click the BkMark button and Use the Set menu option. The default list of known bookmarks is X,Y,Z but you can change that to whatever you want. And you can have known bookmarks in multiple separate categories, if you wish. Just enter the category name before you enter the list of bookmark names and click Store.

The identification of standard (i.e., known) bookmarks is a global setting that will apply to all files. Declaring standard bookmark names does not cause those names to be used in any file… you still set them like before. It is not necessary to make a bookmark known before you declare

it in a document. The only advantage to having known bookmark names is that they will pop up in the menus you see when you use Ctrl and left-click on the BkMark button, as a way to set a known bookmark to the cursor location without having to type in its name.

All about Selections

A "selection" is one or more regions of the file. The highlight you see (for cut/copy/paste) is a selection. The red indications you see with spelling checking is also a selection. Braille2000 also has "named selections" that get saved with the file and whose regions track with editing.

The only selection you can manipulate directly is the Highlight selection. See page 6 for the various ways to set and/or manipulate the Highlight. Indirectly, you can manipulate a selection (Highlight or a named selection) using Adjust / Selection (click the Adjust button). Note that a selection may be composed of multiple isolated regions, as generated by you using the Ctrl key, by the "dtbook" (NIMAS) panel, by Find All, etc. Many operations apply themselves to each selected region of a selection. For example, paragraph style settings apply to all those paragraph that show highlighting.

For certain uses, you may find that the selected regions of a selection are not as broad as you wish. For example, given some selected words (as per Find All), you may want to operate on the entire paragraph at each selected word location. Or having found some element of punctuation (Find All), you may wish to further select the word that it touches, at each location. Such localized expansions of selected regions can be done using Adjust / Selection.

Selection Adjuster

The Selection Adjuster can expand each of the selected regions in a given selection. This operation begins via the Adjust button.

There are four tabs for four composite operations: Expand to whole units, Expand left, Expand right, and Merge segment. In the snapshot you see the Expand to whole units panel. If you select one of the choices (Word, Line, Paragraph, Page) then each selected region has its location and extent "rounded up" to encompass an entire word/line/paragraph/page where it is located. This operation (if used) is done first, before Expand left or Expand right, and if the multiple selected regions then overlap, they are merged together (regardless of whether or not the Merge option is checkmarked).

For example, if Find All had selected two words within the same paragraph, and if you used this panel to expand to whole paragraphs, then that entire paragraph would become select (e.g., it would all become highlighted) and what had been two isolated regions would merge into one region.

With or without Expand to whole units, you can use Expand left and/or Expand right to make each selected region bigger by a given number of a given kind of unit. The "nc" at "by" indicates "no change" and when that is showing, nothing will be expanded. Using the up/down arrows you can change "nc" into a number, 0 or more, for the number of additional units to add
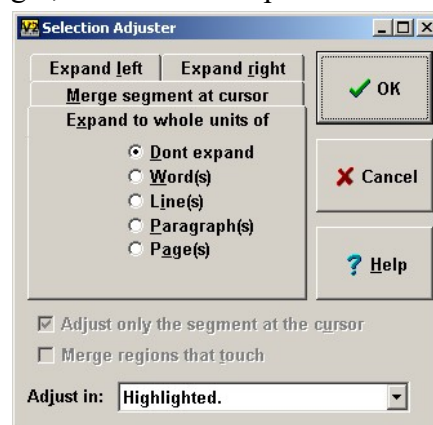
to each selected regions, either to the left or to the right, depending on which tab is active (Expand left is shown above; Expand right looks just the same).

Expand left/Expand right is performed after Expand to whole units has been done. Setting the number of units to 0 often does nothing but it does expand each selected region to the kind of boundary for the unit type that you select. For example, if you have highlighted the "a" in "because" and set "by" to 0 and also select Words, the left side of that selected region would expand (leftward) to the beginning of the word. If you set "by" to 1, it would expand even further leftward to include also the entire word to the left. You can use the 0 count with Expand left and Expand right to get an effect very much like Expand to whole units.

If the selection has multiple segments (multiple regions) and if the cursor is in or at one of those regions, you can use "Adjust only the segment at the cursor" to operate on only that one region. And in such a case, you can use the tab "Merge segment at cursor" to combine that segment with the previous segment and/or the following segment.

The expand operations may cause adjacent selected regions to become overlapping. The "Merge regions that touch" option applies to the operations Expand left and Expand right. (If you use Expand to whole units and an overlap results, such regions are always merged, and the merging takes place before Expand left/Expand right is applied.) When the Merge option is not active, Expand left/Expand right retain the same number of selection regions and will expand each region only as much as could be done without overlapping any adjacent region, i.e., things may not expand as much as requested. When Merge is active, each region expands as requested and if they then overlap, overlapping regions are joined into a single region (i.e., the number of regions in the selection may decrease).

If the file has named selections, you can choose a named selection via "Adjust in" and then the operations apply to that named selection.

Named Selections (Selection Overlays)

Braille2000 allows for optional named selections that are stored in the .B2K file. (To store a named selection you must be working with the .B2K file type.) To manage named selections, click View and choose Selections.

You do not create a named selection directly. You arrange highlighting and then use "Set from highlight" or "Import from highlight" to define a named selection. You can have multiple named selections, each with a unique name in the context of a single file.

Each named selection can show on the screen (or not show) with a color of your choice. By choosing a color that is distinct from the highlight color, you can see what is highlighting and what is from a named selection. Having a named selection showing does not interfere with normal editing or normal uses of highlighting (hence the name "Overlay"). You can change the display mode for each named selection, letting them show (with a given color) or not show. Named selections that are not showing still exist and can be shown later.

47

In the snapshot shown at the right, a named selection called Proofreading is being defined. You might want such a selection if you are doing proofreading for someone else. You can highlight a bunch of words or regions that are problems, and then use the "Import from highlight" button to transfer those selected regions into the Proofreading selection. You can import as often as you wish, building up the Proofreading selection to include all the spots in the file the author should address. If you use "Show with color", then you will see (at all times, until you turn off the Show option) the elements of the Proofreading selection on the screen.

The button "Set from highlight" first empties the selection and then imports from the highlight. It thus makes the named selection identical to what is then highlighted.

The button "Export to highlight" transfers the named selection's regions to the regular highlight, whereupon it can be used with Find or Styles or in other ways.

When a named selection is showing on the screen, you may point the mouse at any one of its selected regions and right-click to see this sub-menu:

If you choose Deselect, that element of the name selection will cease to exist. If you choose Edit you get the Selection Overlay dialog box with the Edit Popups tab active and the panel showing the text associated with the selection element you right-clicked on (the text is initially empty). The snapshot shows that the proofreader has entered a note "Missing G1 indicator" to be applied to selection interval #1 (the one that was right-clicked).

Each region (element) of a named selection may have an optional popup caption that is displayed to the reader when (a) the named selection is showing and (b) the mouse is hovered over the selection (hovering means to put the mouse over the selected region and hold it stationary for a few seconds; do not click any mouse buttons). In this way a named selection can annotate the file in any way you wish, and those selections and annotations are part of the file. If you save the file with such a named selection showing, it will be showing each time the file is opened (until the showing attribute is turned off or the named selection is totally deleted). The task of proofreading someone else's work is an obvious use for this feature, but you can define notes to yourself also, and when you export a named selection to the Highlight, you can do things that operate on each element.
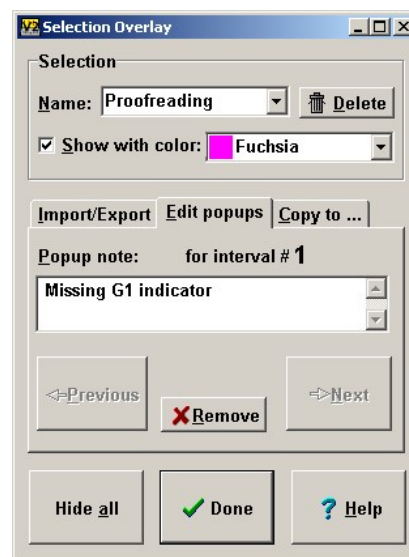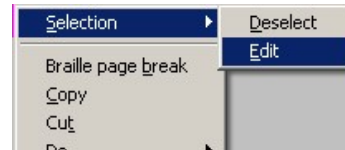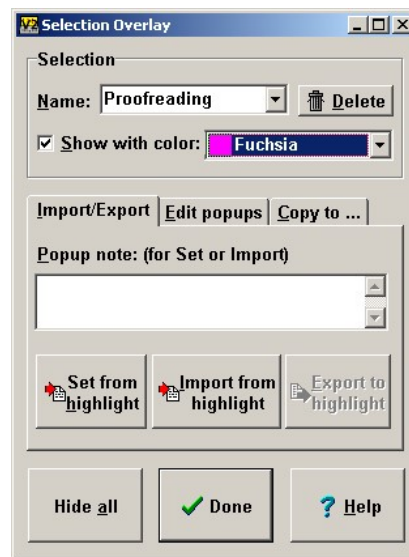
The "Copy to" tab gives you a panel of text harvesting operations. Using it, you can send all of the selected text (that selected by the named selection) to the Clipboard or to a New Workarea in a variety of ways:
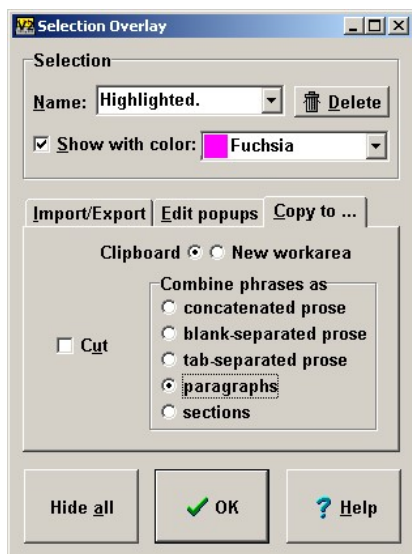
concatenated prose: all selected text is joined into one long paragraph without intervening spaces or other material.

blank-separated prose: all selected text is joined into one long paragraph with blanks between each element

tab-separated prose: all selected text is joined into one long paragraph with tabs separating elements (you can use Ctrl+t to show the tab ruler to position such text)

paragraphs: each element of the selection becomes its own paragraph

sections: each element of the selection becomes its own section (retaining certain page attributes such as page numbers)

If "Cut" is not active, the selected prose remains in the file. If "Cut" is active, the selected prose is removed and after the Copy operation is done the named selection becomes empty.

Quick Code Selection

Setting the braille code for a transcription is often done just once per volume of braille, or not at all if the default code is being used. But when doing foreign language or math (in the embedded Nemeth "Provisional Guidance" approach), code changes may be frequent. To assist in frequent code changes, the system organizes code choices in several ways:

1. **The Default Code**. When you begin a new file, the code is preset to the system default code. You can change this code using the choice box at the right (right-click the Code button to get this dialog box). Select the code and then click "Apply To" and select "System Default".

2. **Code Families**. The code dialog box is organized by families (generally by the organization that is responsible for setting the code rules). There are three families: ABL, BANA, and ICEB. For most transcriptions involving more than a single code, the codes will be from the same family. To change the family, click the "Change" button. Then select the code by name and then select from any options that are offered.
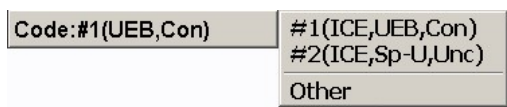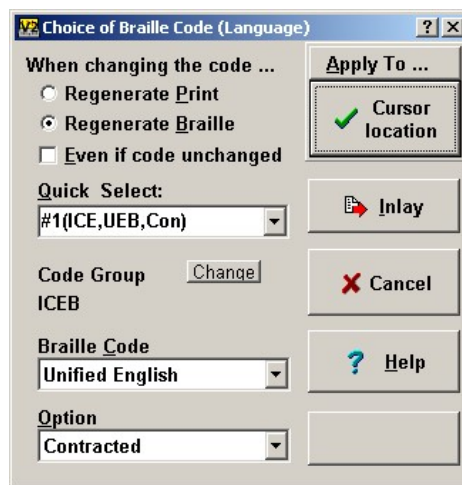
3. **Codes Used in the Current File**. A given braille file may need to use a few different codes, say Spanish and English, or UEB and Nemeth. When you click the Code button (as shown at the left), you will see a menu of the codes already used in your file. To

switch back and forth from code to code, just click the Code button and select the code. File codes are numbered in the order used in the file. The identity of the code is abbreviated to save space . Shown above is ICEB, UEB, Contracted and ICEB, Spanish, Uncontracted.

4. **"Quick" Code Names**. You can define one-word names for codes you use frequently (the name may contain hyphens and underscores but not spaces or punctuation). Each name is a synonym for the code that was selected when the name was defined. Quick names appear first in the menu of Codes (the menu shown in #3 above), and in the Quick Select list (in the Code dialog box).

To define a Quick name, right-click the Code button to display the dialog box (right), select the Code in the usual ways, then click in the text box at Quick Select and type in the Quick name (it will be enclosed by quote marks). Then click "Add Quick Name".
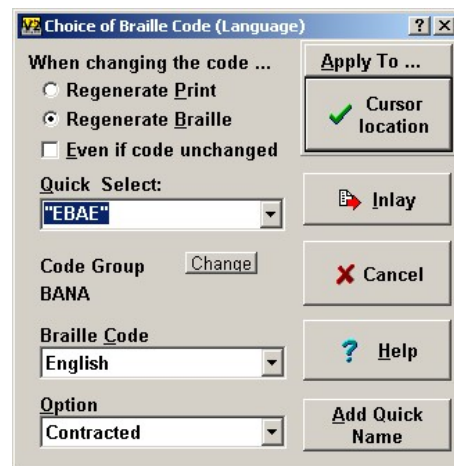
For example, if you do mostly UEB you would have your default Code set to UEB (in the ICEB group). If you occasionally want to work in EBAE, you need to switch groups and although that is not difficult, it takes extra time. The dialog box (above) is shown ready to define "EBAE" as a Quick name. Once this is done, "EBAE" will appear in the menu choices immediately when you left-click the Code button (whether or not EBAE is a code previously used in the file). Quick names are system-level settings: all of the Quick names you define will be available in all of your files.

To remove a Quick Name, display the Code dialog box and select the name using the dropdown arrow button on the Quick Select control. Then click "Remove Name" (this function is handled by the same button that says "Add Quick Name" in other contexts).

Modified Codes
Certain settings (notably the choice of active dictionaries, via the Spelling dialog box) act as attributes on a given code definition. When you set a code attribute (e.g., via Spelling) to a special value, the definition is "modified" from the system standard. For such codes, the word "modified" will show in the Code Choice dialog box, to the right of the words "Braille Code," and legend of the Code button will include the term "mod".
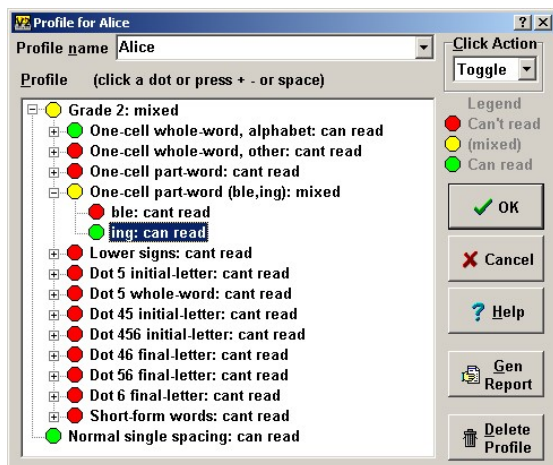
Grade Relaxer

Carried over from V1, there is the re-translation mechanism known as the "Grade Relaxer". This tool takes a contracted document and selectively uncontracts ("relaxes", i.e., spells out) selected elements, thereby reducing ("relaxing") the "grade" of the braille. Element selection is by contraction type. This produces a document tailored to the reading skills of a particular child who is learning contracted braille. This process works on both EBAE and UEB documents.

50

The main dialog box (reached via Grade Relaxer in the "Do" menu group) is shown at the right. You select a profile name (usually named for a particular child) from the list of profiles you have already defined. You adjust the "Text to be relaxed" and select any "Options" and click OK.

The relaxer can also change a single-spaced document into double-spaced (controlled either by the Doublespace checkbox or by the "Normal single spacing" profile element (see below).
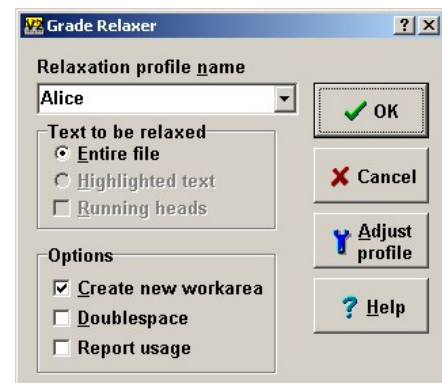
The default behavior is "Create new workarea", i.e., the original file is unchanged and the relaxed copy is written to a new, unnamed workarea, that you can immediately emboss or, if you wish, Save to a new file. If you turn on "Report usage" then a report document (print document) is generated listing each contraction category and the number of times a contraction instance was spelled out.

To create a new profile or further adjust an existing one) you use the tool at the left. It is displayed via the "Adjust profile" button (above dialog box) or by clicking Adjust and selecting "Grade Relaxer Profile". For a new profile, type the name into the Profile name box. For an existing profile, click the drop-down arrow button and select the name you want.

There are 222 contractions listed (including those in EBAE that are not present in UEB, some are listed more than once for different contexts, such as "whole word" vs. "part word"). The list is structured as a hierarchy for ease in navigation. Subordinate entries are shown by clicking a ⊞ box or hidden by clicking a ⊟ box. Clicking a color-dot applies the Click Action to it (toggling between green and red as the default action). Clicking a category dot changes all subordinate elements to the new setting. The selected item can also be adjusted using the keyboard: press spacebar to toggle the color; press – to make it red; press + to make it green. Green denotes those contraction types the child can read (those not to be relaxed). Red denotes those types the child has not learned to read (those instances will be spelled out). The "Gen Report" button creates a hard-copy print listing of the profile.

A green dot for "Normal single spacing" means that normal spacing is OK. A red dot there causes the text to become double-spaced as it is relaxed.

**Analyzing a document with the Grade Relaxer**
You might be interested in knowing the frequency of usage of the various contractions in your document (say for a volume of a textbook). You can get that analysis using the Grade Relaxer if you do the following: prepare a profile for a child who reads no contractions at all (you might call him/her "G1"… everything is red, everything will be spelled out). Then apply that profile to your document and click the checkbox for "Report usage". Let the Grade Relaxer create a new

workarea (that you will then discard). The generated report will tell you how many instances exist in your document, for each contraction type.

Printout Color Control

Screen colors are not necessarily right for printouts. Each kind of printing (the Print As choice) has its own Options reached through the Options button. Within options (just above the page margin settings) you will find the Custom Colors button and you can use it to get the Printout Colors dialog box (right).
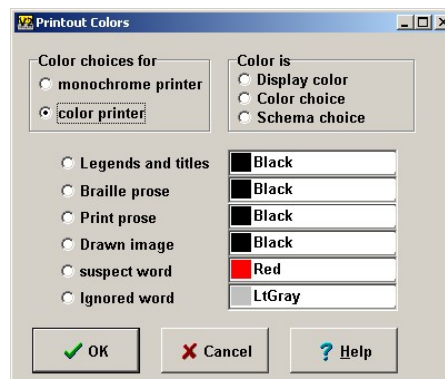
Braille2000 stores two color schemes for printing: monochrome and color. Only one scheme is active at a time (and Braille2000 cannot tell whether you are printing in monochrome on a color printer... you need to select the color scheme that is appropriate).

To set printout colors you should: (1) select the color scheme via "Color choices for"; then (2) select an item (one at a time) that you wish to change; then (3) select the type of color for that item via "Color is"; and (4) choose the color you want that item to print in (a drop-down list of color choices will be available to the right of the selected item).

In the "Color is" selector, "Display color" causes the selected item to be colored like that same item on the screen (the color entry will be that of the screen color). If you select "Schema choice" then the choices for the selected item will be the names of Windows standard colors from your Windows desktop definition.

The color settings are common to all printout options (to all choices of Print As).

Jump To

Jump To Cell in the Search menu and also ctrl+j will display the Jump To… dialog box shown here. This multi-faceted tool lets you automate the positioning of the cursor in a variety of ways. Notice that many sections are grayed out… those elements are inactive until the appropriate request action is selected from one of the radio-box (round bullet choice) items at the left edge of the dialog box.
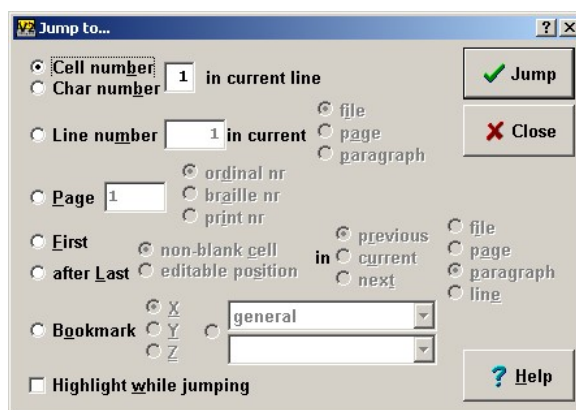
"Cell number", "Char number", … through "Bookmark are mutually exclusive requests. Once you select the main kind of action, its detail elements will become active (un-grayed) and through those choices you can tailor the way you want the cursor to move. Each of the main action request types will be described. The role of "Highlight while jumping" will be described last. The symbol ◘ denotes a fill-in box. The phrases "current line", "current paragraph", "current

page", and "current file" all refer to the element of that kind at the cursor position. The terms previous and next refer to the element ahead of or after the one at the cursor position.

Cell number ◻ in current line
The cursor jumps to the given braille cell position (from the left edge of the page) in the current line.

Char number ◻ in current line
The cursor jumps to the given print character position (from the left edge of the page) in the current line.

Line number ◻ in current file/page/paragraph
The cursor jumps to the first (leftmost) position in the given line relative to the start of the current file/page/paragraph.

Page ◻ ordinal nr/braille nr/print nr
The cursor jumps to the leftmost position of the first line in the given page in the file. The page number can be an ordinal page number (Arabic number denoting page position in the file), or a braille page number (Arabic number with optional section indicator such as t or p), or a print page number (Arabic or Roman number with optional continuation letter). In the case of a print number without a continuation letter, the jump will be to the first page in the file whose corner print number shows that number, with or without a continuation letter.

First non-blank cell/editable position in previous/current/next file/page/paragraph/line
The unit of positioning is file or page or paragraph or line. If you choose previous or next, the jump is to the previous or next unit of the same kind (files have no previous or next unit). The cursor jumps to the first position in the unit that is either a non-blank cell or character or that is a blank space where prose may be inserted (i.e., not page margin or fill).

after Last non-blank cell/editable position in previous/current/next file/page/paragraph/line
The unit of positioning is file or page or paragraph or line. If you choose previous or next, the jump is to the previous or next unit of the same kind (files have no previous or next unit). The cursor jumps to the position in the unit that is just after the final non-blank cell or character in the unit or to the farthest position (past the end of the unit) that is not yet part of the next unit. This would normally be after some virtual blanks after the end of real text, virtual blanks that could become actual blanks if keyboard input were to be done at the new cursor position.

Bookmark X/Y/Z or bookmark category and name
The cursor jumps to the designated bookmark. X/Y/Z are provided for continuity of behavior with Braille2000 V1 (that had only those bookmarks). You can select a bookmark category (upper box) and a bookmark name (lower box) and the jump will go there.

Highlight while jumping
If this checkbox is active then the cells or characters jumped over (jumping either up or down in the prose) will become highlighted.

Note that some effects may require using Jump To more than once. For example if you want to move the cursor to the second line in the next paragraph, you would need to use Jump To to go to the "First editable position in the next paragraph" and then Jump To to "Line number 2 in the current paragraph". Similarly, if you wanted to highlight the first line in the current page, you would need to Jump To "Line number 1 in the current page" and then Jump To "Line number 2 in the current page, while highlighting".

An alternative to jumping while highlighting is to use Menu / Edit / Select and its submenu of selection regions: All, Line, Page, Paragraph, Rest, Word. Another alternative is to highlight using arrow keys and/or PageUp/PageDown (with and without the Ctrl key) while holding down the Shift key. For example, you can move the cursor to the beginning of the current word using ctrl+rightarrow ctrl+leftarrow and then you can highlight the word using shift+ctrl+rightarrow.

"nc"

The values and buttons denoted by the legend "nc" (in various dialog boxes) are settings signifying "no change". The "no change" concept applies to attributes on individual paragraphs. The "nc" setting is used only when you are changing attributes for many paragraphs at a time, as when you have highlighted one or more regions of the file that collectively span multiple paragraphs.

One situation involves various on/off attributes such as "Page turn as top", "Nested list", "Keep together", "Keep with next", "No dbl sp" (in the Paragraph Style dialog box), and "By Page", "Centering full width", "Double spaced", and "Skip page no. line" (in the Layout Control dialog box). These on/off controls are shown as checkboxes: blank indicates "off" and a checkmark (or an "x" in older Windows systems) indicates "on". A normal checkbox has only two states: on and off. The controls just listed behave that way when the dialog box is applied to a single paragraph or when applied to many paragraphs for which the value (of a given attribute) is homogeneous, i.e., the same for all paragraphs. Some attributes may have homogeneous values while others do not.

When one of these dialog boxes is applied to multiple paragraphs with non-homogeneous attribute value, the control for that attribute takes on a third value, namely "nc" (no change). In that case the checkbox has three states: on and off and nc. Clicking the box or pressing the hot-key for that box cycles the setting through all three states. The role of "nc" is explained below.

Other kinds of settings can also have "nc" values. The settings for Adjusted Margins (Left and Right) in the Layout Control dialog box is a number. When applied to multiple paragraphs that have non-homogeneous values for that attribute, "nc" is one of the choices (as if it were a number). This also applies to Blank Lines (Before and After) in the Paragraph Style dialog box.

And then there are the two "nc" buttons in the Paragraph Style dialog box. One of them is for the group "Indent-Runover" and that button sets that group to "nc". The other button is for the group "Additional Properties" and clicking that button sets all the within-group controls to "nc".

The need for "nc"… an example.

Consider the Paragraph Style dialog box. It can manipulate nine different attributes of paragraph style (we consider indent/runover as a single attribute). Consider using the dialog box (directly or by having recorded a Speedbraille™ macro) to make a paragraph centered. This is an interesting situation because there are several variations of "centered" in braille, for example, if just centered there is no need for blank lines before/after, but if a centered heading, then you do want those automated blank lines. And if a heading, you might well want "Keep with next" too.

If you use the dialog box on just a single paragraph (nothing highlighted) and manipulate the settings yourself, the current status of every attribute is shown initially and you simply change those that are not already what you want. The situation gets interesting if you want a Speedbraille™ Key that will make the paragraph become a centered heading (with blank lines) as well as another Speedbraille™ Key that will make the text be centered without affecting its adjacent blank lines. The point is that for some purposes, you would want to change only <u>some</u> of the attributes and leave others alone. This is where "nc" comes in… you set to "nc" those attributes you <u>don't</u> want to change.

So, in this case, we might record a Speedbraille™ Key with all attributes set to "nc" (that's what those two nc buttons are for) but with Indent-Runover set to "Centered". Using this key would set the prose as centered with all other aspects unchanged.

We might also record a Speedbraille™ Key with a full set of attributes (nothing set to "nc") and select "Centered", "1" line before, "1" line after, and "Keep with next". Using this key would set all nine attributes for a given paragraph… it would make it a nice centered heading.

It gets even more interesting when we note that the Paragraph Style dialog box will change settings for all paragraphs in a highlighted portion of the file. And this highlighting need not be contiguous… it could be a selection composed multiple paragraphs scattered throughout the file, having been selected manually or all at once using "The NIMAS Touch"™ automated selections via the dtbook panel (for NIMAS publisher files). Perhaps among all the select paragraphs some already use "Keep with next" and some do not. Do we want to reset those values to all on or all off? Or do we want to maintain that attribute (and perhaps several others) unchanged, i.e., to affect only certain ones of the nine possible attributes. Again, this is where "nc" is used, to designate those things not changed.

When the Paragraph Style dialog box is displayed, the attribute settings reflect the current state of the paragraph or paragraphs that are selected (if nothing is highlighted, it is for the paragraph containing the cursor). For each of the nine attributes, the dialog box will show the value of that attribute if it is homogeneously the same in all selected paragraphs and it will show "nc" if not homogeneous. For those settings where "nc" shows, you can still change that attribute to force a value for all selected paragraphs. If it is a checkbox, you may have to click more than once, because you will cycle the control through "blank", "checked", and "nc". Controls that are not "nc" (a number, or blank, or checkmark) will be asserted for all selected paragraphs. Those that show "nc" will not be asserted (there will be no change to that attribute).

When recording a Speedbraille™ Key (creating a macro), the controls mentioned here (those that have the option of "nc") are passive until changed (by clicking or via their hot-key). If untouched (i.e., if passive), a control is not part of the transaction data generated by the dialog box. It is dialog box transactions that form a Speedbraille™ Key definition. This means that if you want a Speedbraille™ Key to affect some attribute (say "Keep with next") you need to click "Keep with next" at least once while the dialog box is open. This is true even when the dialog box initial setting for "Keep with next" is what you want to set… you still need to change any setting the macro is to affect, even if you have to cycle a control back to the same setting it started with. This notion of a control being passive until once changed is similar to "nc" in its effect.

Print Character Substitutions

You may want to transcribe some print characters in a manner other than what is in print. For example, you may want to treat all the "em dashes" (long dashes) as if short "en dashes". One or more print character substitutions can be performed using the built-in substitutions mechanism. Any selected substitutions apply across an entire print document as it is imported. Braille2000 has, by default, the substitution setting "Short Dash Only" that does exactly what was just mentioned: it replaces em dashes with en dashes, the resulting en dashes become short dashes in braille. For other purposes, you can define your own substitutions, if you need to.

Substitutions apply in print import only. They do not apply to keyboard entry done in the Print View. They do not apply to inserts done using the Insert menu or the Insert button. When you open a source file, there is a dialog box prompt for choosing the desired braille code, and embossing mode, and (now) also the set of substitutions to apply to the print. This choice is called "Print substitutions" and is at the bottom of the Transcription Settings dialog box (shown right). This dialog box appears when opening a source file (a text file, an RTF file, or a NIMAS file). If you do not define your own substitution sets, there are two choices: "None" and "Short Dash Only". The substitutions are made uniformly throughout the print text during the import operation (the original print file remains unchanged).
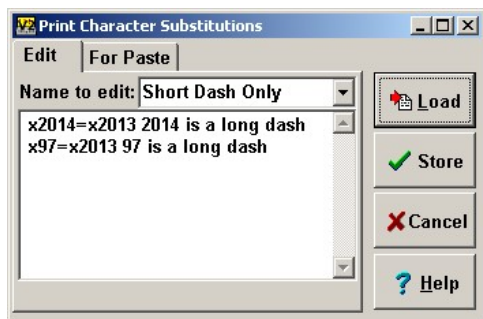
If you need other print substitutions, you can define your own substitutions sets. On each import operation only one substitution set can be active (or none) but it can do several substitutions if you want it to.

You can create and adjust a substitution set using the Print Character Substitutions dialog box (click Adjust / Substitutions). Use the "Edit" tab and the down arrow to select the name of an existing set (Short Dash Only is shown in the image). Click the "Load" button to bring in the current definition, it then appears in the edit panel.

In the edit panel (in the image) you see that this substitution set consists of two substitutions, one for the print character x2014 and the other for the print character x97 (these are explained shortly). Each line of a substitutions set is of the form

[char code]=[one to nine char codes] optional comment phrase

In the first substitution line, the (Unicode) character x2014 (which is the em dash) is to be replaced with the (Unicode) character x2013 (which is the en dash). The text "2014 is a long dash" is a comment that is not processed by Braille2000. If any comment is written it must be set off from the last character code by at least one blank. The second line says that character x97 (which is an upper-ASCII code for em dash used in RTF files) is to be replaced by (Unicode)

character x2013 (the en dash), that is, <u>both</u> x2014 and x97 (should they exist in the prose) get replaced by x2013.

The notation xNumber is a character specification using hexadecimal notation. I.e., 2014 is the hexadecimal Unicode value for the em dash. An equivalent way to specify a character is by its decimal Unicode value in the form uNumber, as in u8212 (which is the same number expressed as a decimal value… the em dash again). The Unicode character numbers span the printable ASCII portion (u32 to u126, or equivalently x20 to x7E) and the so-called "upper ASCII" characters (u128 to u255, or equivalently x80 to xFF). Because most substitutions will be for Unicode characters not found on the keyboard, the xNumber or uNumber notations are the only ways to refer to arbitrary Unicode characters.

The leading character code is the character that is to be replaced. The replacement follows the equals sign and can be a string of one to nine character codes (if more than one, they are written all touching each other). For example (this is a silly example, you wouldn't likely want it), if you wanted every short dash to be replaced by -- (two hyphens), you could write the following line (note that u8211 is the en dash and u45 (same as ASCII 45) is the hyphen. You could also use the appropriate xNumber notations or a mix of xNumber and uNumber notations.

> u8211=u45u45 short dash to two hyphens

If you create your own substitution set you must give it a unique name. If you create or edit a substitutions set, you need to Store it using the Store button.

Note that you can replace only individual print characters (not multi-character sequences), and you can replace a character only with one to nine other print characters (not with braille cells).

For Paste

The Transcription Settings dialog box is used for Open Source File (when a new document is being created). For Paste and Import (i.e., into an existing document), that dialog box does not show. In those cases, the substitution set (if any) that is used is the one set by the "For Paste" tab. Select the substitution name you want and then click the Store button. This setting will persist, for Paste and Import (print only) operations, until changed. You can reach this panel (For Paste) also via right click to the Paste button and selecting "Set Substitutions". If you want substitutions, set the substitutions first and then perform Paste.
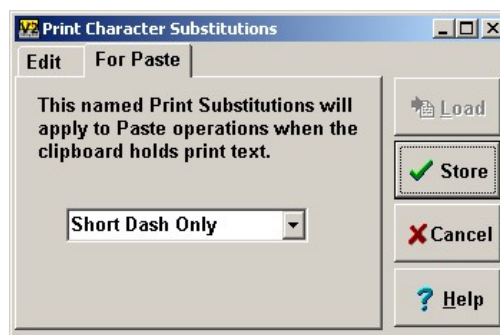
Page Numbering

Most aspects of automatic page numbering in V2 are the same as in V1, but there are some important changes, and there are some important tips for using page numbering. Page numbering is one aspect of page layout… in braille that means stealing part of the page area for other than running prose. The page numbering mechanism plants auto-updating page numbers in the corners of braille pages as well as inserts page turn indicator lines in the middle of pages. The behavior is by way of actions provided by strategically located control annotations that exist in

three classes: Numbering, Braille Sequence (braillenum), and Print Sequence (printnum). All three kinds work together to determine how page numbers are generated.

Numbering Annotation

The Numbering annotation specifies the locations on the page where page numbers will show. There are settings for literary style (braille page numbers in the upper right, not normally used), textbook style (braille page numbers in the lower right; print reference numbers in the upper right), music (print reference numbers in the upper left; braille page numbers in the upper right), and unnumbered (no corner page numbers but print page transition lines, i.e., a page turn indicator line without any page number). There is also a customizable setting (linear form) for use when transcribing specifically for single-line braille note-taker devices. The numbering style is usually homogeneous throughout a transcription, and thus the typical file has only one Numbering annotation, located at the top of the file. It is possible to change the numbering style for some parts of the document, but usually such changes are signs of errors and are unwanted. See below for fix-up techniques.

BrailleNum Annotation

The job of enumerating the sheets of paper is handled by the braille page number. A BrailleNum annotation is planted on the first page and specifies the starting number. The number is incremented for subsequent pages. There are one to three braille number sequences in a transcription, often occurring all in the same file. There are "transcriber pages", "preliminary pages", and "text" pages. There will be a BrailleNum annotation hung on the first page in each sequence, with the typical transcriber page number being t1; the typical starting preliminary page number is p1, and the typical starting text page number is 1.

PrintNum Annotation

Print reference numbers show the flow of braille prose relative to the pages in the print copy of the book. The transition to a new page of print can occur anywhere in the prose, and in textbook format, that location is shown by a horizontal line of dashes, the page turn indicator line. A PrintNum annotation is planted at the first word on the new print page. The system generates a page turn indicator line unless the transition is at the top of a braille page or too close to the bottom of the page (in such situations, the print number in the corner of the page is all that changes). In a sequence of braille pages following a given PrintNum annotation, page reference numbers receive a continuation letter prefix automatically: a single PrintNum annotation handles the transition and all subsequent continuation pages, up to the next PrintNum annotation. The number of PrintNum annotations in a transcription is the same as the number of print pages in the book. A handy menu feature is "Next Print Page" that marks the next page transition (at the location of the cursor) and posts a print reference one higher than the previous one.

Page Numbering Dialog Box

All three kinds of page numbering control annotations are manipulated by the same dialog box. When you click Do and Page Numbering, you
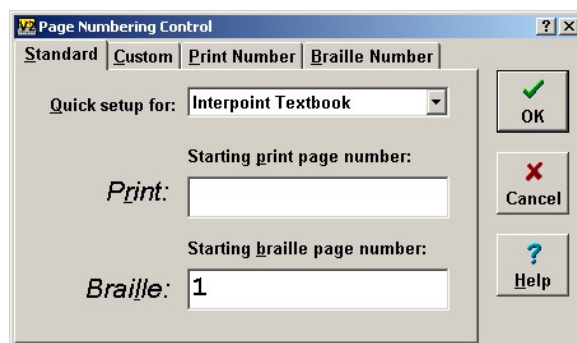
get the Page Numbering Dialog, as shown here.

The default layout is Textbook format. If you have set your file's Embossing Mode to Interpoint, the default choice becomes Interpoint Textbook (see Interpoint Page Numbering below). When working in the Print View, the dialog shows print notations, as shown here. When working in the Braille View, it shows braille dots. For Braille numbers, you can enter a plain number (e.g., 1) or a type letter and number, such as p1 for preliminary pages and t1 for transcriber pages. You can assert page numbering for your file whenever you want (you don't have to do it first or last)… page numbers and page turn indicator lines will appear dynamically as you add to your transcription. For most needs, the Standard tab and its print and braille inputs are all you need. The Print Number and Braille Number tabs handle just those aspects, respectively, with additional options (but there are no additional braille number options).

Print Number Extras

There are times when you need to show a range of print pages, such as 4-7 as shown here. One use is to account for spanned pages that lack transcribed prose, say text on page 3 but not on pages 4, 5, and 6. Just enter the range of pages included. When you use Next Print Page, the next number will be 8. Any continuation pages will start with the number a7.
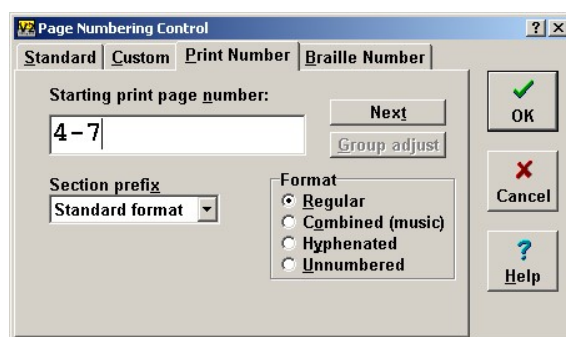
If however you transcribe a multi-page tableau that has a layout that ignores page boundaries (usually just two facing print pages), you write that range of pages the same way but you need to select "hyphenated" under Format. Next Print Page will again advance to page 8 but any continuation pages for 4-7 will be shown as a4-7, b4-7, etc.

If you choose the Unnumbered option, the print reference number you give will be stored in the file (potential later use) but it will not show. An unnumbered page turn indicator line will show, even at the top and bottom of the braille page. Choosing the Unnumbered print number format changes Quick Setup For to "Unnumbered Pages" (you can use that setting equivalently).

The choice "Combined (music)" generates combined print reference numbers, as needed for music transcriptions.

Some publications you may transcribe have chapter-based page numbers such as 1-1, 1-2, 1-3, or even Unit 1-1, Unit 1-2, Unit 1-3. Depending on the salience of such notations, you may or may not want the extra stuff in the print reference number. When you desire greater format control for print numbers, select "Use format below" under Section Prefix. Put the number part of the print reference number in the "Starting print page number" box as for ordinary page numbers. Put the extra stuff into the Section Prefix box and indicate, using the "Insert format item" button, where in the text the continuation letter and number parts should appear. The choices for format items are "Continuation Letter" and "Active Number" and these positions are marked by special symbols:
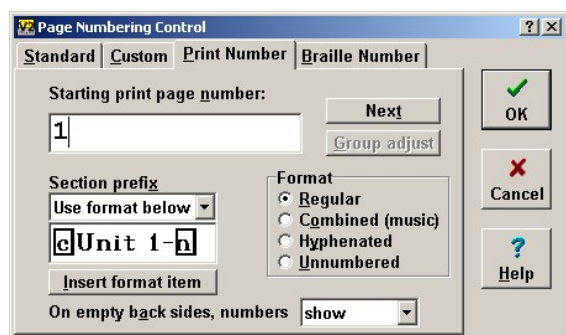
a c in a box [c] where the continuation latter will go, or an n in a box [n] where the number part will go. For example, using this control you can determine whether the continuation letter appears before "Unit" or after "Unit" or after the hyphen.

The Section Prefix choice "Use Prefix Below" is a simplified version of the above. No explicit format tokens ([c] or [n]) are used. You enter the print number prefix and the system presumes the continuation letter precedes it and that the active number follows it.

If your transcription is for interpoint embossing (as set using Adjust / Document), the choice box at the bottom of the Print Number panel will show, letting you choose between "show", "don't show", and "don't count" as the way to handle print number that would normally appear on an empty back side page of an interpoint page pair.

Interpoint Page Numbering

Interpoint (two-sided) embossing is not much different in page layout than single-side embossing. If you have an interpoint-capable embosser, Braille2000 will let you emboss on both sides regardless of the page numbering settings or the file's embossing mode (embossing mode set to interpoint does set the default embossing technique as well as enable features that distinguish between front and back braille pages).

For reasons unknown, there is no standard arrangement of page numbers for interpoint transcriptions. The useful choices are: (a) to arrange the page numbers just as they are for single-side embossing, or (b) to save a wee bit of space by suppressing the braille page number on the back sides, the understanding being that the braille number provides for re-collating the pages should a volume become unbound—clearly the front and back of the same sheet cannot become separated or jumbled, and thus the braille number on the back side is unnecessary. The agency requesting the transcription or the end user can choose how to adjust page numbering for interpoint. Some agencies may choose to suppress the print reference number on the back side. Some agencies may choose to suppress all page numbers on back sides that are otherwise empty.
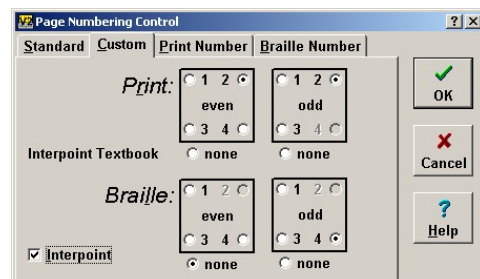
In Braille2000, Quick Setup For Interpoint Textbook does suppress braille page numbers on back-side pages. This arrangement is fully adjustable.

The Custom tab shows a panel of squares representing different page behaviors. The illustration is for a document whose embossing mode is interpoint. When the embossing mode is single-side, half of the boxes vanish because there is no longer a distinction between "front" and "back". By way of the bullets, the locations (which corner) for print and for braille page numbers is determined. The choice "none" suppresses the print number (it exists but does not show). In the default arrangement for Interpoint Textbook, the choice for braille page number position for even pages is "none". If you change that selection to corner 4, you have the same settings as Textbook. If you need an unusual layout, you can move the locations of print and braille page numbers as you wish.

The Power of Highlighting

For page numbering operating via the Page Numbering dialog box, you need not highlight any prose before using the various functions. The page numbers you request will take effect at the document location of the cursor at the time you click "OK" in the dialog box. Depending on the settings you use, one action may generate some or all three kinds of annotations that control page layout for page numbering. It can sometimes happen that you get (by mistake) excess control annotations in your document. If that happens, highlighting, done just before using Page Numbering can help you get this going right again. The behavior of Page Numbering is more powerful when used while a region of the file is highlighted.

The essence of the extra power is that, within a highlighted region, any conflicting annotation are cleaned away before the new settings take place… and unknown and unwanted annotations are eliminated. The portion you should highlight relates directly to the way page numbering works.

If the layout of the page numbers is not consistent through your file, or if page numbers simply will not appear in some part of the file, the problem can be bad Numbering annotations. Normally the page number layout is homogeneous throughout your file, and so to clean that up, you should do the following:

1. use ctrl+a (select all) to highlight everything.
2. (while highlighted) click Do / Page Numbering and change Quick Setup to NoNumbers; click OK
3. (while highlighted) click Do / Page Numbering again and change Quick Setup to whatever kind of layout you want (e.g., Textbook); click OK
4. press ESC to remove the highlight
Because of the highlighting, Numbering will be uniform throughout your file

If the behavior of braille page numbers is not consistent through a given section of your file, you should do the following.

1. highlight all the pages in the section you want to repair, i.e., highlight all text pages, or all preliminary pages, or all transcriber pages.
2. (while highlighted) click Do / Page Numbering and fill in only the correct starting braille page number, i.e., 1 or p1 or t1; click OK
3. press ESC to remove the highlight.

If all sections are bad, you can use the procedure three times, each section done separately.

If the behavior of print page numbers is not consistent through the prose from a given print page, you should do the following.

1. highlight **exactly** the prose from the given print page
2. (while highlighted) click Do / Page Numbering and fill in only the correct print page number; click OK
3. press ESC to remove the highlight

Do this procedure for each print page (separately) that has page numbering problems.

Using Unnumbered Pages

Early literacy braille may involve double spacing and simplified textbook page numbers called "Unnumbered Pages". In this mode, print page transitions (also called "turnovers") are indicated by a full-width line of (36) cells, without any number. Braille page numbering remains as with other transcriptions. Although print page numbers do not show, **you should transcribe the work as normal textbook format**, specifying the print numbers in the normal way… in fact the recommended technique is to complete the transcription using Textbook page numbering, and then when the file is done, highlight everything (ctrl+a) and use Do / Page Numbering to set Quick Setup For to "Unnumbered Pages" (the page numbers will then vanish but the turnover locations will endure).

Most early literacy braille is embossed single-side. In the event you wish to emboss the material as interpoint, you can suppress braille page numbers on even (back-side) pages by clicking the Custom tab in the page numbering dialog box and making sure there is a checkmark at "Interpoint" and selecting "none" for the even braille pages.

Using Linear Form

The Linear Form style of page numbering is intended for transcriptions explicitly prepared for delivery to a single-line braille note-taker device. In this mode, there are no corner page numbers (neither braille nor print) but each turnover (print page change) is indicated by a line that you can control, the idea being to make it distinctive so that the note-taker's search function can be used to find successive turnover locations. As with Unnumbered Pages, you should transcribe the work as textbook format (and it is recommended to complete the file using textbook page numbering and then, when it is finished, to highlight everything (ctrl+a) and use Do / Page Numbering to set Quick Setup For to "Linear Form". Via the Custom tab you should declare the nature of the turnover lines. If you know the line length of the refreshable braille display, it is recommended to set the document page width to that value. It is also recommended to avoid using running heads and footers.

When doing Linear Form page numbering, the Custom tab is used to declare the composition of the turnover line (the separator line otherwise called the "page turn indicator line")… it does not have to be (36)s followed by the print number, although that can be achieved. (If you do not set any customizations, Linear Form is like Unnumbered Pages without braille page numbers.

In the Custom panel, shown here, you can specify braille sequences for the left lead-in and the right lead-out. For each lead-in/out you can optionally stretch the sequence to fill the line. "Include print number" determines whether the print number appears in the turnover line or not. Note that the standard page turn indicator line can be achieved with the lead-in set to (36), lead-out empty, lead-in stretched, print number selected.

As shown, "pg " will precede the print number after a string of (36) cells stretched to fill the line. Notice that the lead-in ends with a blank.
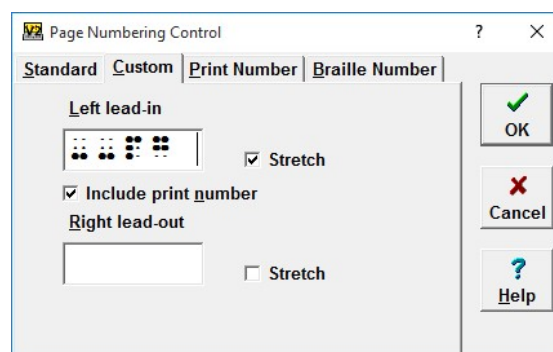
When the lead-in/out is stretched, the cell of the longest run (i.e., the cell that occurs the greatest number of times adjacently) is replicated to lengthen the lead-in/out as needed. As shown, the (36) cell occurs the most adjacently and thus the turnover line will consist of lots of (36) followed by "pg " followed by the page number. If you removed the checkmark on "Include print number" then "pg " would end the line (and you would probably not put in the trailing blank in that case).

If you removed the checkmark on "Stretch" then the turnover line would be (36)(36)(1234)(1245) ( ) followed by the page number, beginning in cell 1 (a short line).

If you use only the lead-out box, then the page number (if any) would appear first followed by the given lead-out material (stretched or not, as desired).

If both lead-in and lead-out are given and both are stretched, the print number will appear (more or less) centered, with both lead-in/out stretched in equal amounts to fill the line.

A variety of turnover notations can be achieved using these settings.

Creating a Specials Symbols List

You can read about standard Special Symbols lists in 2016 BANA Formats. There you will find the specific heading and the layout and ordering specifications. A special symbols list is normally found at the front of each braille volume, and it normally lists only those special symbols actually used in that single volume.

Braille2000 can generate the special symbols list for you, in just seconds. It does this when you import into your document (when it is essentially complete, having in it all the symbols it is ever going to have), a special symbols list template file. You can create such a file yourself, but you can also use the one supplied with Braille2000. It is called "UEB Special Symbols.RTF" and is located in the program installation folder, normally at C:\Program Files\Braille2000\BRL2000B.

The special symbols list file is an RTF print file. It contains the heading SPECIAL SYMBOLS USED IN THIS VOLUME followed by many lines of symbol definition prototypes, each in a special and exact format such as this line

y ⠈⠯   Ampersand

Each template line **must** begin with a single letter (one of the letters bhirRsuy) followed by one space followed by the braille symbol being defined (in a simbraille font) followed by one space and followed by the symbol description. You cannot use tabs in a template line. You do not include the dot locator for mention.

The special symbols template contains all the symbols you might ever use, listed in the proper order (braille order). The section of adjacent template lines must be introduced with the directive %symbolslist and it must be ended with the directive %symbolslist=off (unless nothing follows it in the file).

When template lines are imported into an existing braille file, those lines bounded by %symbolslist to %symbolslist=off are processed by checking the target document for the given braille symbol used in the role as indicated by the leading character. The roles are:

b	braille without any print equivalent
h	hyphen
i	indicator
r	letter or contraction
R	prefix for letter or contraction (modifier)
s	space or whitespace
u	punctuation
y	symbol (other than punctuation)

(It can be that the same cells, in different contexts, have different roles, and hence different descriptions. The notion of role also prevents cells, such as those that can be either a contraction or a punctuation symbol (depending on context) from being falsely recognized. (Fortunately, Braille2000 knows all the roles of standard braille expressions.)

When processing %symbolslist material, a quick search is done in the target document for each braille symbol occurring in the given role. If such use is found in the target document, the line is altered to remove the leading role letter and put in its place the dot locator for mention. If no such use is found for the symbol, its template line is deleted. The result is a special symbols list containing just those template lines with symbols actually used in your transcription.

In preparing the template file, you must use the proper role letter, otherwise the symbol in your target file will not be recognized and the template line will not appear in the resulting list, even though the braille cells exist in your transcription.

If you find errors in UEB Special Symbols.RTF or if you want to add template lines or if you want to remove template lines, you can edit that file using any word processor and save your own customized version of it. You use this file merely by importing it into an already-transcribed document that is open in Braille2000.

Working With Multi-code Documents

Some transcriptions cannot be done in a single braille code. Some foreign language study books need a mix of English and some other (uncontracted) language code. The "Provisional Guidance" notion for math uses a mix of UEB and (uncontracted) Nemeth. Depending on the needs of the braille readers you serve, you may still need to produce materials in EBAE or Nemeth.

The Code button in the control panel is the starting place for selecting the braille code you need to use. When Braille2000 starts, the empty workarea is assigned the system default code (how to set this code choice is covered below). The legend of the Code button is a dynamic statement of the code that is used in the document at the location of the cursor: if you move the cursor through a document with multiple braille codes, the legend on the Code button will change. In an initial empty workarea, the legend will indicate the default code, in abbreviated terms. It will show something like #1(UEB,Con) and it means that where the cursor is located, the #1 code is active and that it is contracted UEB. Every document uses one or more braille codes and they are given ordinal reference numbers for ease of use. The first (sometimes only) code is #1.

When you left-click the Code button, you get a popup menu listing the Codes already in use in the document, plus the choice "Other". If you are doing a book that is a mix of English and Spanish, then the choice menu will show UEB and Spanish and Other (once the transcription is under way and you have already added Spanish and UEB material). The notion here is that in a multi-code transcription, you are most likely to switch to a code you have used before. For a new document, the Code choice menu normally shows only the default code choice plus Other. You can configure Braille2000 so this choice menu always has several code choices that you choose (see below).

If you right-click the Code button, or if you left-click and select "Other", you get the "Choice of Braille Code" dialog box by which you can select a variety of codes and perform some code management tasks including re-translating all or part of your document into another code. In this dialog box, the code choices appearing in the popup choice menu are available via the selection box called Quick Select (making a selection via Quick Select is much the same as doling left-click on the Code button and selecting from the popup list). However, Quick Select can do more.

If you click in the text of Quick Select (rather than on its dropdown arrow button), the text becomes highlighted and you can enter replacement text from the keyboard. You can put in a name you choose (but composed only of letters, numbers, hyphen, and the underscore symbol). Then if you click the "Add Quick Name" button, the name you set becomes a synonym for the code shown in the dialog box (you should adjust the settings to indicate the code you want, before you type the name into Quick Select). Names you add this way will always appear (a) in the choices menu (via left-click to Code), and (b) in the Quick Select choice list. Braille codes in Braille2000 are grouped by primary code authority agency and it takes an extra step (clicking the "Change" button) to select a code from another group. Thus it is (intentionally) not as easy to select UEB and EBAE on a frequent basis. If you need to do that, give one or the other or both of them quick names.

Re-translating the Document

Braille2000 contains both forward (print-to-braille) and backwards (braille-to-print) translators in each of the codes available in the code dialog box. Because of this, Braille2000 can offer both braille and print equivalent texts (when saved, Braille2000 documents store braille, the print is derived each time the file is opened). Using derived print notation, it is possible to re-translate that to braille, in a code that is different from the original. The quality of such re-translation depends on the quality of the derived print, and in this regard it must be noted that EBAE contains fundamental ambiguities that can, in some uses, make back-translating to print uncertain, and thus sometimes wrong when compared to the original print document. When re-translating EBAE, you should proofread the result.

The notion of braille code is precisely the bi-directional equivalence of utterances in braille with those in print. If you change the code on existing prose, then one or the other of the braille and print manifestations **must** change. When both exist a priori, as they do in Braille2000, you get to choose which one (braille or print) is to be regenerated, the point being that the form not regenerated anchors the other and does not change. Sometimes you set the code in preparation for text input… if at the current end of text in the document, then there is nothing already there to be re-translated… the setting affects the way new material is handled when entered on the keyboard or pasted in from the clipboard.

Re-translation can be applied to all or just the highlighted part of a file. It might typically be done to change between contracted and uncontracted forms, or to change between uncontracted Spanish (following World Braille Usage) to UEB (using three-cell accented vowels). Such changes can be done either direction. You can also convert between UEB and EBAE, but remember that UEB to EBAE is usually accurate (because UEB is essentially unambiguous) while EBAE to UEB can be problematic (because of ambiguities in the definition of EBAE).

These re-translations are all handled the same way, via the "Choice of Braille Code" reached via right-click to the Code button. All you have to do is to select what part of the file you want to re-translate (e.g., by highlighting it), and then select the new code, and select either Regenerate Print or Regenerate Braille at the top of the dialog box. Obviously when you need a different braille code, you should select Regenerate Braille, and this is the default setting. But there are other times, when the braille itself is already what you want but when it was provided in a manner in which the system does not know the correct code setting, such as when opening a "dumb" braille file such as a BRF type. If Braille2000 does not have the right code setting, its associated print data will be flawed. When you use the code dialog box to set things straight, you need to select Regenerate Print, keeping the braille as is.

If your document contains a mix of codes, for example contracted UEB with many uses of uncontracted Nemeth, it can be very tedious to change all code regions one at a time, as when you wish to change the UEB to uncontracted while not disturbing the Nemeth parts. You can handle this by highlighting all the UEB parts of the file, at the same time, and then changing the code. The trick is knowing how to highlight by code designation. You do this by a right-click to the Code button, then select the code you want to highlight via Quick Select, and then click the "Highlight This Code" button. If you then change the code, the change will apply to just the highlighted regions and not the other regions.

Using Find and Replace and Associated Functions

In Braille2000 there is one common dialog box to handle a number of related functions:
    Find: search for a given string (braille or print)
    Find All: search for all such strings (braille or print) and form a multi-highlight
    Find and Replace: search for a given string (braille or print) and replace it with another
    Replace: replace the highlighted prose (replace each segment of a multi-highlight)
    Find and Insert Before: search for a given string and prepend inserted prose (braille or print)
    Insert Before: prepend prose to the highlighted segment or every segment in a multi-highlight
    Find and Insert After: search for a given string and append inserted prose (braille or print)
    Insert After: append prose to the highlighted segment or every segment in a multi-highlight
    Find and Extend: search for a given string then search for the ending string and highlight the
        range
    Extend: search from the beginning of a highlighted segment for the ending string and
        highlight the range or so extend every segment in a multi-highlight

The Find/Replace dialog box has tabs for Find, Replace, Insert Before, Insert After, and Extend. With each tab there is a separate edit box that can contains ASCII-braille, simbraille, or print text (an option for regular expressions is not yet implemented). For each tab, the panel shown has

action buttons, each of which will close the dialog box. Replace, Insert Before, Insert After, and Extend have different behaviors depending on the edit box of the Find tab, as explained below.

Find

Ctrl+f presents the Find/Replace dialog box with the Find tab selected. Put the string you want to find in the text box. The buttons [A] [B] [P] and [R] set the text box to ASCII-braille, (sim)braille, print, and regular-expressions, respectively (support for regular expressions (i.e., wild-card searching) is pending). If Find Words is checked, then a matched string must be situated surrounded by whitespace. If a single segment of the file is highlighted when the dialog box is displayed (and the segment is not too long), that prose becomes the initial setting of the text box.
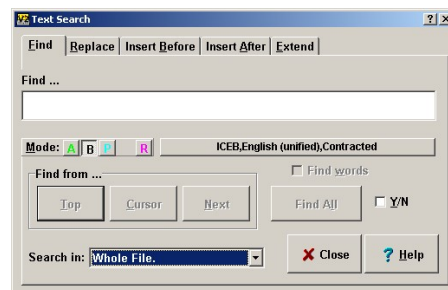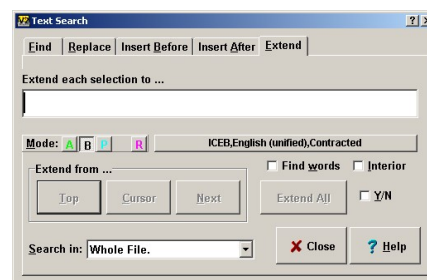
The buttons Top, Cursor, and Next initiate the search for one more matching string in the file, starting from the top of the file, the current cursor position, or just after the last match position, respectively. The Search In selecting may say Whole File, or Highlighted, or the name of a named selection and constrains the part of the file that is searched. If Find All is clicked, then the search through the given region finds all matching strings and leaves them all highlighted. If Y/N is checked, then at each potential match, the matching string is shown highlighted and the legend "Y N Esc" appears in the upper left corner of the screen: you press y to record the find, n to ignore the find, or Esc to stop the process (Y/N is typically used with Find All). When Find completes, the item(s) found remain highlighted. If Find All is not used, then the search process may be continued by pressing Ctrl+n. If the text box is empty, no action buttons are enabled.

Replace

Ctrl+r presents the Find/Replace dialog box with the Replace tab selected. The behavior of the action buttons depends on the prose in the Find text box. If the Find text box is empty, Replace requires highlighted prose (one or more segments) and operates to replace each entire segment with the Replace box text. If the Find text box is not empty, then the action buttons perform a search for the Find text and then that text is replaced by the Replace text.
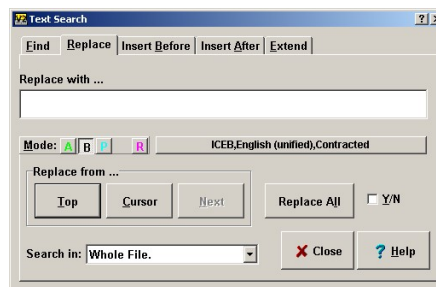
The buttons Top, Cursor, and Next initiate the search for one more matching string in the file, starting from the top of the file, the current cursor position, or just after the last match position, respectively. The Search In selecting may say Whole File, or Highlighted, or the name of a named selection and constrains the part of the file that is searched. If Replace All is clicked, then the search through the given region finds all matching strings and replaces each of them with the replacement text. If Y/N is checked, then at each potential match, the matching string is shown highlighted and the legend "Y N Esc" appears in the upper left corner of the screen: you press y to replace the material, n to ignore the find, or Esc to stop the process (Y/N is typically used with Replace All). When Replace completes, the highlight is removed unless the Find text box was empty and then the highlighting remains. If Replace All is not used, then the search and replace process may be continued by pressing Ctrl+n.

Insert Before

Search / Find Text Before presents the Find/Replace dialog box with the Insert Before tab selected. The behavior of the action buttons depends on the prose in the Find text box in much the same ways as when doing Replace, the difference being that the Insert text is prepended to existing prose rather than replace it. If the Find text box is empty, Insert Before requires highlighted prose (one or more segments) and operates to prepend to each entire segment the Insert box text. If the Find text box is not empty, then the action buttons perform a search for the Find text and then that text is prepended with the Insert text.
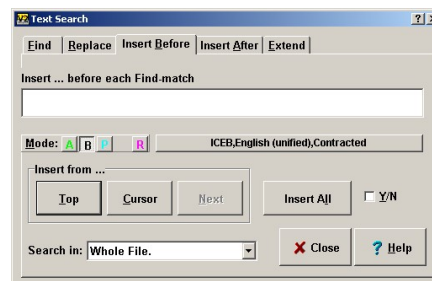
The buttons Top, Cursor, and Next initiate the search for one more matching string in the file, starting from the top of the file, the current cursor position, or just after the last match position, respectively. The Search In selecting may say Whole File, or Highlighted, or the name of a named selection and constrains the part of the file that is searched. If Insert All is clicked, then the search through the given region finds all matching strings and prepends each of them with the inserted text. If Y/N is checked, then at each potential match, the matching string is shown highlighted and the legend "Y N Esc" appears in the upper left corner of the screen: you press y to prepend the material, n to ignore the find, or Esc to stop the process (Y/N is typically used with Insert All). When Insert completes, the highlight is removed unless the Find text box was empty and then the highlighting remains. If Insert All is not used, then the search and prepend process may be continued by pressing Ctrl+n.

Insert After

Search / Find Text After presents the Find/Replace dialog box with the Insert After tab selected. The behavior of the action buttons depends on the prose in the Find text box in much the same ways as when doing Replace, the difference being that the Insert text is appended to existing prose rather than replace it. If the Find text box is empty, Insert After requires highlighted prose (one or more segments) and operates to append to each entire segment the Insert box text. If the Find text box is not empty, then the action buttons perform a search for the Find text and then that text is appended with the Insert text.
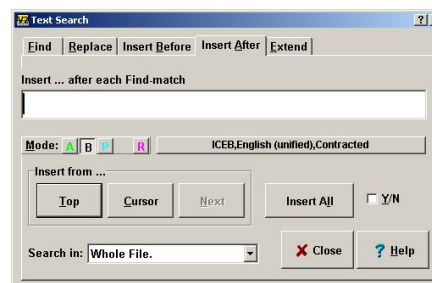
The buttons Top, Cursor, and Next initiate the search for one more matching string in the file, starting from the top of the file, the current cursor position, or just after the last match position, respectively. The Search In selecting may say Whole File, or Highlighted, or the name of a named selection and constrains the part of the file that is searched. If Insert All is clicked, then the search through the given region finds all matching strings and appends each of them with the inserted text. If Y/N is checked, then at each potential match, the matching string is shown highlighted and the legend "Y N Esc" appears in the upper left corner of the screen: you press y to append the material, n to ignore the find, or Esc to stop the process (Y/N is typically used with Insert All). When Insert completes, the highlight is removed unless the Find text box was empty and then the highlighting remains.. If Insert All is not used, then the search and append process may be continued by pressing Ctrl+n.

Extend

Search / Extend Text presents the Find/Replace dialog box
with the Extend tab selected. The behavior of the action
buttons depends on the prose in the Find text box in much the
same ways as when doing Replace, the difference being that
Extend creates a highlighted segment rather than change any
prose. If the Find text box is empty, Extend requires
highlighted prose (one or more segments) and operates to
lengthen each segment to the following string matching the
Extend box text. The search for the extended end of the segment is not allowed to overlap the
next highlighted segment: if there is no match of the Extend string before the beginning of the
next segment, the highlight for the segment being extended is removed. If the Find text box is
not empty, then the action buttons perform a search for the Find text and then the text
immediately following the matching Find text is searched for a match to the Extend string, and if
such a string is found, the prose from the beginning of the Find text to the end of the Extend text
is highlighted, except that if Interior is checked, then only the interior is highlighted (i.e., not the
Find text nor the Extend text strings themselves).

The buttons Top, Cursor, and Next initiate the search for one more matching string in the file,
starting from the top of the file, the current cursor position, or just after the last match position,
respectively. The Search In selecting may say Whole File, or Highlighted, or the name of a
named selection and constrains the part of the file that is searched. If Extend All is clicked, then
the search through the given region finds all matching strings and extends each of them to the
position of subsequent matching Extend text, as modified by the Interior checkbox. If Y/N is
checked, then at each potential match, the matching string is shown highlighted and the legend
"Y N Esc" appears in the upper left corner of the screen: you press y to extend the highlight, n to
remove that highlight, or Esc to stop the process (Y/N is typically used with Extend All). When
Extend completes, the highlighting remains. If Extend All is not used, then the search and extend
process may be continued by pressing Ctrl+n.

Find/Extend with Interior checked can be used to find and highlight all the TNs in a volume. You
put the beginning indicator in the Find text box and the ending indicator in the Extend text box
and then click Extend All.

Initial Views

Braille2000 is a multi-document tool, meaning that you can open many braille files at the same
time. And each file can be viewed in various ways as selected by the four buttons A B P S
located in the Control Panel or via the Menu and View and View As (Alt v v). You can also get a
split screen view if you select a lower-half-window view by clicking a different view button
while holding down the Ctrl key on the keyboard.

The View Selector will show a named tab for each file you have open but it will also have a tab
for each additional view you have created for a file. For example, you can have a tab for the
braille view of the file and another tab for the print view of that file (as well as use the four
buttons to switch any given view to a different mode). Supplemental views of a file are obtained

using View and Add View. Each view of a file performs editing on the same file but each has its own scroll position and its own cursor as well as (possibly) different display mode. Supplemental views are good to see the file in different ways (e.g., print vs. braille) but two views of the same mode (e.g., braille and braille) are useful because of independent cursors: you can have one view showing the table of contents while the other view is scrolled down in the file to show what a contents entry refers to–it's the same file, just multiple access windows to it. You can have as many views (in whatever display mode you wish) to the same file as you wish. And with multiple monitors, the views can (via View and Windowize) be positioned on different monitors.

Automatic View Restoration

Beginning with release 2.275, Braille2000 will record the views in use when Braille2000 closes and will restore those views automatically when you start it later. The system records the display mode(s) used and the cursor and scroll positions of each view, so it will be just the same at a later launch of Braille2000. As you would expect, restoring a view means re-opening the file being edited, and thus if you move, rename, delete, or unplug the drive for a file, the view cannot be restored automatically (you would need to re-open the file yourself). With this in mind, you might want to plug in any removable media (e.g., flash drive) before you launch Braille2000 if you want files you were working on to have their view(s) restored.

The views that get restored automatically are those in effect when Braille2000 closes. The files shown in those views should have been saved so that they are no longer "modified" and so that Braille2000 will close *without* reminding you that the file needs to be saved: save all the files, then close Braille2000. This usage is needed because files that you Save only when reminded to do so during the Close operation are both Saved and Closed (assuming you click "Yes" to save the file). This means that the views for such files are already closed as Braille2000 terminates, and are thus not eligible for automatic restoration. (There is nothing "wrong" with saving your work only when reminded during Close, it just doesn't lead to view restoration.)

Similarly, if you don't want a file view restored the next time Braille2000 launches, then Close that file before you close Braille2000. You can close a view using the lower "X" button in the upper-right corner of the window, or by a right-click to its View Selector tab, or by Ctrl+F4, or as well as via the View menu. There may be an instance when you have closed Braille2000 with active views (that will auto-restore) but you don't want that to happen (because it takes extra time, or because the file causes a problem, or because you want to paste fresh material into an empty workarea). In that event, if you hold down the Shift key during launch, automatic view restoration will be bypassed and Braille2000 will launch with an empty workarea.

Creating New Documents Using Paste

A common approach to creating a new braille document is to Copy print material (e.g., from WORD) to the Clipboard and to launch Braille2000 and Paste it in. This works well because Braille2000 launches with an empty workarea ready to accept Paste material, at least when there are no latent document views being restored: when you close Braille2000 with active views, they get restored at the next launch and the empty workarea is not provided. If you need to do Paste in that situation, there are several options:
       1. You can launch Braille2000 and close all the restored views (an empty workarea will then be provided).
       2. You can launch Braille2000 while holding down Shift.

3. You can launch Braille2000 and then click New and Workarea.

Working With Paragraph Styles

Probably the main task a transcriber performs is managing the format of the braille document. Because a braille page contains relatively little prose, all space is precious and at the same time the layout needs to be consistent for tactile reading.

Page layout is the result of two processes: applying paragraph style and applying page layout controls. This section is about paragraph style. (Page layout control is asserted via various annotations for things like page numbering, line numbering, layout control, page control, and other features found in the "Do" menu.)

Paragraph styles get asserted in multiple ways: via left-click to the Style button and using the style popup menu, via right-click to the Style button and using the Paragraph Style dialog box, via Do / Change Format or Adjust / Paragraph Style in the menu, and by file import mechanisms including "%-codes". Directly or indirectly these assertions relate to the Paragraph Style dialog box shown here.
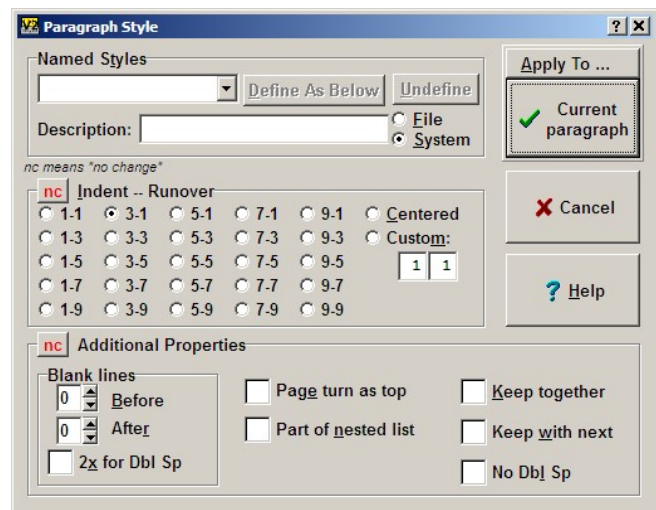
The notions of braille paragraph style revolve around the concepts of indent (cell number where the first line begins) and runover (cell number where all other paragraph lines begin). Some uses of prose, such as centered headings, need an automatic blank line before and/or after the prose. That is indicated by the numbers in the Blank Lines box. For managing the flow of prose across page boundaries, there are the settings of Keep Together and Keep With Next. Each of the Additional Properties is discussed below.

The Apply To button determines the scope of the asserted paragraph style. If nothing is highlighted when the dialog box is used, the button reads Current Paragraph (as shown). If text is highlighted when the dialog box is used, the button reads Highlighted Text. When you click that button, all paragraphs with some highlighted text will be style-adjusted. (Note that in Braille2000 you can highlight text as multiple disjoint passages, all of which will be affected.) By clicking on the words "Apply To" you can select other scopes, including All Of File, Rest Of File, and System Default (the style applied to the first paragraph of a new empty file).

The alternative style mechanisms of Braille2000 (e.g., the popup style menu) let you select a style by way of a named "style macro" (a concept similar to Speedbraille™ Keys). The system contains a number of built-in style macros with mostly intuitive names, such as block1, block2, indent1, list2, head2, etc. Those styles are what you see when you left-click the Style button and select a style from the popup menu. Those names are also what you see when you click the down-arrow button in the Names Styles box in the upper-left part of the dialog box shown above. If you select a style name, the settings in the dialog box will indicate exactly how that style is defined, and if needed, you can change those settings (caution advised).

The named styles mechanism is that of a dialog box macro, i.e., it is handled like a Speedbraille™ Key for a particular use of the Paragraph Style dialog box. And because it is implemented like a Speedbraille™ Key, user-defined named styles show in the list of Speedbraille™ Key and can have either System or File availability. Furthermore, you can use the Speedbraille™ Keys dialog box to assign a named style to a trigger key or to give a conventional Speedbraille™ Key a style name instead of a trigger key (but only if its definition consists of just the use of the Paragraph Style dialog).

Using the Paragraph Style Dialog Box

This section discusses each of the buttons and settings found in the Paragraph Style dialog box. The dialog box is shown by: right-click to the Style button, or by selecting Adjust / Paragraph Style or Do / Change Format in the menu (via the Menu or Adjust or Do buttons).

Style Settings

Indent-Runover This group of radio buttons selects commonly used indent/runover pairs for formatting the paragraph. For indent/runover values other than the common pairs, choose Custom and set the indent and runover values numerically in the boxes provided. For centered prose, choose Centered. The "nc" button activates the "no change" mode for indent/runover (see the discussion of "nc" earlier).

Additional Properties This group of settings adjusts miscellaneous formatting properties associated with the paragraph style. The "nc" button sets all properties to the "nc" value (see the discussion of "nc" earlier). The checkbox controls in this section are three-state controls, showing blank (off), checkmark (on), or nc (no change). You may need to click the control more than once to cycle to the setting you need.

Blank Lines This pair of controls establishes the number of automatic blank lines needed before and/or after the prose of the paragraph. Adjust the number of lines using the arrow buttons.

2x For Dbl Sp This checkbox controls whether or not the number of blank lines should double when the layout is set for double-spacing.

Page Turn As Top This checkbox (formerly denoted "Heading") controls whether or not the style treats an immediately preceding page turn indicator line as if it were the top page boundary, i.e., when active no automatic blank lines are inserted between the prose and the immediately preceding page turn indicator line.

Part Of Nested List This checkbox determines whether or not the paragraph bases its runover value on the gestalt nature of the surrounding nested list (in a nested list, all paragraphs have the same runover behavior with the cell number being two plus the largest indent setting on paragraphs in the list).

Keep Together This checkbox determines whether or not the multi-line prose of the paragraph can straddle a page boundary. When active, if the prose would otherwise straddle the page boundary, a page break (type K) is inserted at the start of the paragraph, thus promoting it to the

top of the next page. If the paragraph is so large that it cannot fit on any single page, Keep Together is ignored. This same concept exists in WORD and other document processors and is imported (where it has been used) from RTF data.

Keep With Next This checkbox determines whether or not this paragraph can sit at the bottom of a page (i.e., having a page boundary between it and the subsequent prose). When active, if the subsequent paragraph would otherwise start on the next page, a page break (type K) is inserted at the start of the paragraph thus promoting it to the top of the next page. If the paragraph is so large that it fills or overflows an entire age, Keep With Next is ignored. This same concept exists in WORD and other document processors and is imported (where it has been used) from RTF data. Keep With Next has accumulative effects when used for several adjacent paragraph and/or with Keep Together.

No Dbl Sp This checkbox causes the paragraph to ignore the double-spaced layout setting (it remains single spaced).

Defining Named Styles

Named Styles This group of buttons and data is used to define and manage named styles. It can also be used to alter the behavior of built-in styles (caution is advised). By clicking the down-arrow button, you can see a list of all the known style names (both built-in and user-defined). When you select a name, the other controls of the dialog box are configured according to that definition… by inspecting the various settings you can see how the style is defined. If you want to create a named style of your own, using a name not in the list, you can click in the name box (a blinking cursor should appear) and then you can type your own style name (only simple names are allowed to be style names).

Define As Below When you click this button, the style name that is showing is defined (or redefined) according to the settings of the dialog box. The name will have System or File availability according to the selection of File or System. File-based names are available only for the file for which they are defined. System-based names are available for all files. If you are redefining a built-in style, you probably want to select System.

Undefine When you click this button, the previously defined style name is removed. If the name matches that of a built-in style, the built-in definition is reestablished.

Description Enter an optional remark to remind yourself what the style is designed to do.

Paragraph Styles as Speedbraille™ Macros

Internally, a named style is a Paragraph Style dialog box macro, i.e., a Speedbraille™ definition. The Paragraph Style dialog box can show you the list of all available named style and just like with Speedbraille™ Keys, the available items may depend on the file you are editing (there are both System and File definitions). Named styles will also show in the list of Speedbraille™ Keys (but only those you have defined; unaltered built-in styles do not show in that list).

In the list of Speedbraille™ Keys, a named style has a style name instead of a trigger key. Whereas a Speedbraille™ definition can do many things together, a named style is limited to performing but one act: the assertion of style settings. In this way, a named style is a limited use

of the Speedbraille™ mechanism. Via the Speedbraille™ list, you can replace a style name with a trigger key, turning it into a Speedbraille™ Key. The reverse (changing a Speedbraille™ Key into a named style is possible only if the Speedbraille™ definition is just a single style change (more complex macros may not become named styles).

Like Speedbraille™ definitions, a copy of the definitions is automatically included in every automated braille file (.B2K or .BML type) you save and can be transferred between users by exchanging files.

Named Styles As %Directives

When importing Rich Text prose (RTF data), you may elect to insert %directives to assert format choices (including paragraph style) and other behaviors. For named styles with names that are distinct from standard %directives (defined earlier in this document), you can assert the named style by using %*stylename* in the RTF document. For example, if you have defined the named style "Exercise" you can format RTF paragraphs that way by using %Exercise as a directive. %directives are not case-sensitive, so you could also use %exercise to select your Exercise named style.

By design, many built-in paragraph styles are named to match the corresponding %directive. The list of such styles is shown here. Beginning with version 2.276 these %directives are coupled to the corresponding named style and you now have the ability to redefine the style should the need arise.

| Block |
| Centered |
| Head |
| Indent |
| List |
| Note |
| Subhead |
| Toc |

When you need a built-in style to behave differently, identify the style name (or names) and use the Paragraph Style dialog box to define your specialized version of the style. Set the availability to System, or set it to File and make sure the revised style is already known in your braille file by the time you import RTF material in which you use that name in a %directive.

For example, if you want %centerd to denote a centered heading (and not merely centered prose), you could redefine the named style called "centered". You would set the number of blank lines before and after and perhaps turn on Keep Together and Keep With Next. This would change the behavior of %centered as well as the behavior of the "centered" item in the popup style menu (that you see when you left-click the Style button).

If you do not want to change the behavior of the popup Style menu, you might instead use the Paragraph Style dialog box to define a new style name, say "ch", and give it the settings you want. Then you could use %ch in your RTF files to get that effect.

Remember that your own named styles (those you have defined) will be included in the Speedbraille™ Keys listing (because of the intimate relationship between named styles and Speedbraille™ macros). You can use View / Speedbraille™ Keys to see what you have beyond the built-in styles.

INDEX

%

%directives. *See also* formatting layout
    code naming, 30
    defaults to contracted, 29
    explanation of, 22
    language options, 29
    list of special in braille, 24
    macro options, 30
    special in braille, 24

A

ambig trans
    explanation of, 10
Annotation, Braillenum, 58
Annotation, Numbering, 58
Annotation, PrintNum, 58
annotations
    change from inception, 13
    change starting here, 13
    changing use of, 12
    control of, 11
    control of visual appearance, 3
    dialogue box, 12
    explanation of, 3, 11
    hover for description, 11
    inspect function of, 12
    moving with cursor, 12
    not used for document control, 11
    remove, 13
    show/not show, 11

B

blank page
    inserting, 16
bookmark function, 14
bookmarks
    explanation of, 45
    instructions, 45
boxing lines, 14
    types of, 14

C

clipboard, 7
code dialogue box

inlay explanation, 44
    math, 44
code selection, 49
    default, 49
    families, 49
    inlay, 45
    naming, 50
    used in current file, 49
Code selection
    quick select, 65
color scheme
    default not light cyan, 2
    gray, 2
    light blue, 2
    red, 2
    uses of, 2
    white, 2
    yellow. *See*
compatibility of version 2
    with Windows 8 and 10, 1
control key
    and arrow key behavior, 3
control panel. *See also* specific buttons
    button explanations, 13
    explanation of, 13
    hiding. *See*
    NIMAS files, 13
copy. *See* cut, copy, paste
Ctrl key
    actions, 8
    braille actions, 9
cursor
    home and end key behavior, 3
    inter-cell gap, 3
    movement of positions, 3
    pg up and pg down behavior, 4
    two positons between cells, 1
    within yellow and gray areas, 2
cut, copy, paste, 7
    change cursor location, 8
    cursor location, 8
    paragraphs, 8
    phrases, 8

D

dash, long vs short, 56