

# Table of Contents

Eclipse 7.0.....	5
Connection Magic.....	5
Dialog for Connection Magic settings.....	5
Import Bridge Mobile annotations as proofreaders marks.....	5
Status bar server connection indicator.....	5
Add message box to connection dialog.....	6
Add page/line number to connection magic editor list text.....	6
Connection Magic user properties.....	6
Save session passwords for users to be able to look up later.....	6
Remember "publicly visible" option permanently in user settings.....	6
Some way to "hand off" output session ownership between reporters.....	6
Report Bridge Mobile users to Eclipse reporters.....	7
Feature to kick users out of sessions.....	7
Warning messages for shared editors when host disconnects or lags.....	7
Separate colors for chat users.....	8
"Find" button on collaboration dialog.....	8
Private messaging for Connection Magic.....	8
All document info should be transmitted through sharing.....	8
Production.....	9
Multi-page enhanced.....	9
Multi-page presets.....	9
Multi-page document overrides.....	10
Multi-page header/footer now support job variables.....	12
Multi-page full-sized cover page added.....	12
PDF output added.....	14
PDF digital signatures.....	14
PDF concordance indexing.....	14
PDF auto-indexing.....	15
PDF embedded exhibit files.....	16
PDF embedded exhibit / auto-index linking.....	16
PDF encryption and protection.....	17
PDF/A-1b compliance.....	18
Transcript delivery dialog to integrate print/PDF/ASCII/etc. into one step.....	18
Continuous statistics updating.....	20
variable for note file creation date/time in form fields.....	21
EXAUTO for numbers or letters auto-incrementing.....	21
Multi-page should be able to index speaker names.....	21
Add ASCII import to dictionary conversion format list.....	21
ASCII import should deal with timecodes.....	21
ASCII import should have options for ignoring numbered header lines.....	21
Allow student version to create RTF with almost nothing in it.....	21
Allow users to modify the index font size.....	21
Auto-reconciliation for ALPHASORT vs. NUMSORT.....	22
Some way to export the notes from an .ecl file.....	22
Send e-mail option in Create ASCII dialog.....	22
Allow non-student users to turn on "School mode" to try extra features.....	22
Allow students to output to Bridge Mobile without a license upgrade.....	22

Auto-Brief.....	23
Auto-brief max words setting.....	23
Allow AB background window to change color.....	23
Word-wrap briefs in the pop-up window to a maximum width.....	23
If user asks for another brief use alternate forcing theory.....	23
Auto-brief settings dialog should have buttons to link to related programming tab items.....	24
Option for number of times word/phrase must be written to produce a brief.....	24
Right click context menu on auto-briefs: reject/new/open.....	24
Auto-brief plural detection should reverse spelling adjustments and work for requests.....	24
ABNEW should put the new suggestion on the same list it deleted it from.....	24
{ABDELETE} feature to remove a brief defined as the next stroke written.....	24
{ABSELECT} now allows accepting a specific suggested brief.....	25
{ABCHANGE} requests new steno for a specific brief.....	25
Translation.....	26
Add Passport Touch writer menu item defaulting to 115K baud.....	26
Passport shadows option.....	26
Button in Translation Magic settings to link to programming tab items.....	26
Continuous redacted text mode for translation.....	26
Translation Magic rule for 'contains steno'.....	27
Spell mode abort strokes should be user-definable.....	27
Insert missing BY line needs to differentiate between Q and A by lines.....	27
A way to have close paren not trigger a paragraph punctuation.....	27
A locked conflict should not display as unselected when chosen by "pick 1st choice.".....	28
Conflict AI-sharing.....	28
Make Stenograph pedal equivalent to + key.....	29
Add primary language selector to settings.....	29
Autopunctuation shouldn't try to change the punctuation unless the sentence is relatively simple.....	29
Delayed number triggers.....	29
Date formats should allow for an optional automatic trailing comma.....	30
Option to put integral prefix/suffix in untran color.....	30
Button near integral pre/suf translation setting to open programming tab.....	30
Realtime.....	31
Auto-block feature for boilerplate text.....	31
Deny saving should be permanent setting.....	32
Bridge filename should transmit periodically.....	32
Bridge refresh ON should be the default.....	32
Add a setting for using the Windows default audio setting instead of using a specific device.....	32
Send script line dictionary entry that doesn't require a macro.....	32
Make "start from which stroke" realtime message optional.....	33
Division interval should be able to expire when no writing is occurring.....	33
Output COM port recovery.....	33
Username/password and reconnection automation for TCP-IP connections such as Speche.....	33
Add date preset drop-down list under the Realtime filename template field.....	33
Dictionaries.....	34
Analyze documents dictionary report of text that appears with different steno.....	34
Add phrases to analysis.txt using auto-brief rules.....	34
Multi-select on analyze documents feature.....	34
Ignore variations in analyze documents.....	34
Don't apply suffixes in analyze documents feature if not in English.....	34

Dictionary merge should have 'keep all' option.....	35
Dictionary print should keep steno/text column order to match display.....	35
Dictionary print should use dictionary name as document name.....	35
Dictionary print should allow variable font size.....	35
Dictionary spell check should look for substrings.....	35
Editing.....	36
Find speaker function should work for by lines.....	36
Find feature should optionally wrap around to top of document and continue to starting point.....	36
Find feature should optionally remember last search string.....	36
Find feature remembers anywhere/word/etc. if selected using radio buttons.....	36
Allow user to hit "Shift+Ctrl+," in the globaling window to get {,?}.....	36
Multi-layer jump to last edit point.....	36
'Selective' global prompting for each instance like a search and replace.....	36
Indexing wildcards should treat lockspaces and spaces the same way.....	37
Permanent bookmarks.....	37
Add blank field dialog options to add font/attributes after creating field.....	37
Add a way to apply a macro to a list of files.....	37
Optionally allow Auto-globaling misstrokes to use slop stroke syntax instead.....	38
Partial auto-replacement.....	38
Undo function for join paragraphs should restore paragraph attributes in the process.....	38
Reading a file into another should create paragraph types that don't exist.....	38
Allow copy current/master to do labels and fonts based on paragraph remappings.....	39
Toggling open and close double/single/double quotes.....	39
Alt P Q D should work with one-space punctuation if declared in metadictionary.....	39
Hyphen in editing should not add a lockspace if removed in metadictionary.....	39
Backspace should remove all punctuation declared in metadictionary.....	39
Allow directional quotes in editing if declared in metadictionary.....	40
Down-cap in editing if there is no cap command for metadictionary.....	40
Wrappable non-space characters like - and /.....	40
User interface.....	41
Extended edit controls in dialogs.....	41
Build index item should remember previous entries in drop boxes.....	41
Keystroke dialog droplist should contain number keypad combinations.....	41
Job report should be a non-modal dialog box.....	41
Job report allows sorting of the commands by text or chronological.....	41
Allow super/sub definition to specify percentage of main font size.....	41
Use steno dialog in slop strokes table.....	42
Alphabetize programming tab items.....	42
Right-clicking on a toolbar should take user to customize toolbars.....	42
Edit box should support reverse / and ?.....	42
Special characters dialog should contain a font button.....	42
Internet browse menu item function should handle URL encoding.....	42
Spell check.....	43
Permanent watchwords list.....	43
List errors should be a non-modal dialog.....	43
List errors should use marked block only, if applicable.....	43
Add Google lookup button to spell check dialog.....	43
Add Google suggestions to Global Magic and spell check lists.....	43
Spell check-as-you-type for dialog edit boxes.....	44

Speakers.....	45
By lines should use colloquy colors.....	45
Ability to change speaker names in blocks.....	45
Speaker table changes should trigger refreshes.....	45
Speaker table entries in dictionaries should work with “apply globals”.....	45
Form field / speaker table relationships added.....	45
Audio.....	46
Audio level button functionality for newer versions of Windows.....	46
Multi-channel audio recording/playback.....	46
New multi-channel recording player interface.....	49
File management.....	50
New file dialog created from scratch.....	50
File manager should have tabs for all folders specified in File Locations.....	52
Add {DESKTOP} file locations (like {DOC} for documents).....	52
'New folder' functionality for the file locations feature.....	52
Button for Programming/file locations in User settings/User/Advanced.....	52
Transcripts should remember creation software version number.....	52
File manager should offer to close all appropriate file types.....	53
Translate dialog should contextually decide to use “open” or “save” dialog.....	53
File locking feature to prevent future changes.....	53
When working on a read-only document continue to remind users not to make edits .....	53
Captioning.....	54
SMI format conversion .....	54
Apply timecode offset to .scc file export.....	54
External COM port pass-through enhancements.....	54
Script lines copied to text document should use current timecode.....	55
Add UI for CC block/pass settings.....	55
Speech.....	56
When using speech TM should automatically be disabled.....	56
Disable number formatting .....	56
Use dictation mode .....	56
Use realtime timecodes .....	56
Put in time adjustment for key/voice duplexing.....	56
Speech should be able to start session with mic off.....	57
SpeechText output should disallow content on selected paragraphs.....	57
Presence of SpeechText.dix file should determine whether Raw Text export contains linefeeds.....	57

# Eclipse 7.0

## Connection Magic

### *Dialog for Connection Magic settings*

Under User settings / User / Advanced there is a button for “Connection Magic” which opens up a dialog. The “Server URL” setting is the same setting that would normally require editing the Eclipse.ini file and changing the “Server=etc.” line. Note that after changing this line, it is necessary to exit and re-run Eclipse in order for the change to work. The dialog will automatically prompt for that.

The most common settings for the server are connect.eclipsecat.com:4020 and LAN:4020, so those can be easily selected from a droplist without having to type them in.

Remember, the software will auto-detect the appropriate server to use. You only need to change this if you have Internet access available but nevertheless want to use Connection Magic features such as Bridge Mobile or shared editing only on the LAN.

### *Import Bridge Mobile annotations as proofreaders marks*

Also in the Connection Magic settings are settings for the URL where the Bridge server posts should be sent, and the Bridge Mobile cloud document service e-mail and password.

After setting up a Bridge Mobile cloud user, you can store Bridge Mobile documents permanently. In order to link up Bridge Mobile and Eclipse, it's necessary to enter that same e-mail and password in Eclipse.

Once that's set up, you can go to Tools / Convert / Text files / To / Bridge Mobile proofreading. This will convert the document and transmit it to the cloud file server.

Once this is done, you can go to Bridge Mobile and open the transcripts dialog. Under “unfiled” the document exported from Eclipse should appear.

You can go through the document looking for errors and mark them using either quick marks or issue codes. Since issue codes are user-definable, you could use issue codes such as “spelling” or “punctuation” or etc. You can also write notes.

After proofreading, close the document. Back in Eclipse, go to Tools / Convert / Text files / From / Bridge Mobile proofreading. That will open the original file and will turn all of the issue codes and notes into comment lines at the appropriate points in the document. These comment lines will include page/line references, the issue code, and the notes.

### *Status bar server connection indicator*

In the status bar, there is now a SRV indicator. If that is on, then the connection to a Connection Magic server is active.

If not, it will be blank. This is similar to the status bar indicators for HYP for hyperkeys, INS/OVR for insert/overtyping mode, etc.

### ***Add message box to connection dialog***

Under the Tools / Connection Magic / View connections dialog, there is now an edit box at the bottom of the dialog. You can type a message into that dialog and hit [enter] to send a message, rather than having to use the send message function.

### ***Add page/line number to connection magic editor list text***

When using the shared document feature of Connection Magic, it is now possible to see what page/line number each editor is on just by glancing at the list of editors.

### ***Connection Magic user properties***

The View connections dialog now has a "properties" button that allows you to select a user who is connected to a collaboration session and hit Properties to see the properties of that user.

### ***Save session passwords for users to be able to look up later***

If you start a Bridge Mobile or shared editing session and you forget the password, you can now look it up by going into the View connections dialog, select your user and hit the Properties button. The user properties dialog now has a "Password" button that you can use to look up the password.

### ***Remember "publicly visible" option permanently in user settings***

Most users usually either prefer their Connection Magic sessions to always or never be public, so this option is now remembered permanently.

### ***Some way to "hand off" output session ownership between reporters***

This is still experimental, so I'm asking users for feedback on how it should work. With this version, it is now possible for multiple Eclipse reporters to join a Bridge Mobile output session.

Here's how it works: One user starts a Bridge Mobile output session and tells the other reporter(s) what the name and password is. They can then start a realtime session, and when the output session name appears, they can type in the exact same session name and password.

It will still report the "session name in use" error message, but it will offer the option: "Press OK to join as an alternate provider." If the additional reporter(s) hit OK, and if they have the password right, they will be hooked up to the same output session.

Here's what this means: Any reporter can write, and it will go out to the Bridge Mobile session. That will allow reporters to quickly and easily hand off from one to the other when job sharing

without even waiting for a break. Just don't write at the same time or the viewers might see a mashup of two reporters writing every other word!

You will note that on the View connections/Users/Properties there is a button for "Give control." Originally, it was planned that the reporter doing the writing would have to hit this button in order to give control to the other user who could then start writing. After trying it out, it actually seemed like an unnecessary extra step, so unless anyone sees any huge problem with the way it's working now, I'll remove that button.

So here's the question: Is it better to allow the logged-in reporters to just start writing when its their turn with no extra steps, or is it preferable to force the current writer to have to explicitly give control to another reporter in order to avoid accidentally writing over each other.

### ***Report Bridge Mobile users to Eclipse reporters***

The newest version of Bridge Mobile and the Connection Magic server software will now report the Bridge users who are currently connected to the realtime session.

Note that this requires the Connection Magic Local server software 1.1.0, or the version of the server-side software at [connect.eclipsecat.com:4001](http://connect.eclipsecat.com:4001), along with the Bridge Mobile 1.1.0 version at [connect.eclipsecat.com:8881](http://connect.eclipsecat.com:8881)

This will report the username and e-mail that is typed into the Bridge Mobile "info" tab of the settings dialog. The information about the user will appear on the "view connections" dialog in Eclipse.

### ***Feature to kick users out of sessions***

In the "View connections" dialog, it is possible to see who is connected to a session. Historically, this could only see editors connected to shared document sessions and Bridge desktop users connected to Bridge sessions.

With this version, it is now also possible to see Bridge Mobile users and other Eclipse reporters as alternate realtime providers connected to Bridge sessions.

No matter who else is connected, it is possible for the session owner to select a user from the connected users list, hit the Properties button, then press the "Remove" button to remove that person from the session. They will immediately be disconnected and will not be permitted back into the session.

### ***Warning messages for shared editors when host disconnects or lags***

Some reporter/editor teams working with shared documents have reported issues when connections are slow, particularly with scopist edits being undone, since the connection isn't fast enough to keep the changes current and it eventually has to reconcile back to the reporter's version, which will be slightly older, by seconds or even minutes.

This version will now warn the editors when the reporter's system is either lagging behind or is disconnected. When scopists receive this message, they should stop editing that document temporarily since those edits may not stick. (They can edit other documents while waiting, if

desired.) Once the reporter's system catches up, another message will appear letting the scopists know that it is safe to resume editing.

### ***Separate colors for chat users***

Under Tools / Connection Magic / View connections / user Properties, there is a “Color” button that allows you to select a color for that user's chat window messages.

This selection will also determine the color of the editor's name in the flag that appears on the editing screen.

### ***“Find” button on collaboration dialog***

Under Tools / Connection Magic / View connections / user Properties, there is a “Find” button that will find the selected user in the current document. That allows you to find another user in the team editing process.

### ***Private messaging for Connection Magic***

Under Tools / Connection Magic / View connections / user Properties, there is a text box and a “Send” button that will send a private message just to the selected user. This will show up with “(Private from [username])” in the message on the recipient's screen.

### ***All document info should be transmitted through sharing.***

This originally started out as several separate requests: The shared document feature should transfer the speaker table. It should transfer the list of job dictionaries. It should transfer the job variables. It should transfer the paragraph and document settings. Etc., etc.

Every one of these requests has now been addressed with a single monster feature: The entire non-content portion of the documents is now transmitted over the wire. Document settings, Paragraph settings, job variables, speaker table, job dictionary list, document statistics, everything.

Keep in mind that this does open up the possibility of collisions. If one editor modifies the document settings, they're immediately transmitted to everybody. If two users edit the document settings at the same time, these changes could cross in transit and only one change would be preserved during reconciliation. Overall, this shouldn't be a huge issue, since changes to the settings are extremely rare compared to other edits.

# Production

## ***Multi-page enhanced***

The multi-page output has been dramatically enhanced to allow both greater flexibility and ease of use. It has also undergone some tuning in the appearance of the index and other miscellaneous items. The following items will enumerate the specifics.

## ***Multi-page presets***

Multi-page can be daunting to set up because there are so many options. Market research shows that nearly everyone, no matter what multi-page software they use, now conforms to a nearly universal standard: Four pages per sheet, a four-part box drawn around all of the pages, a binding margin on the left, custom headers on the top and bottom, and just the 25 numbered lines and page numbers within the boxes.

In order to make it easier to set up, The multi-page dialog now offers three styles, labeled "Style 1, Style 2 and Style3" in the drop-list at the top of the multi-page dialog. Select a style and hit the "preview" button to see a sample of a transcript creating using the selected style.

If you wish to use that style, hit the "Select" button and all of the multi-page settings will be changed so that your multi-page printout will look like the sample.

If you wish to customize from there, it will be much easier to start with a pre-defined style and tweak a few items than to start from scratch and try to set everything manually.

The three styles are, in summary:

- Style 1: Boxed pages, Two-line header and footer, Arial and Courier fonts, Bold answers
- Style 2: Boxed pages, one-line header and footer, Times New Roman and Lucida Console fonts
- Style 3: Individual page boxes, individual headers/footers on each page from the original transcript, Arial fonts

Customization hint 1: Fonts and font sizes are the easiest things to change. Pick a style based on the overall shape of the layout and change the fonts from there. Changing the number of header lines, or the layout of the boxes, is trickier since you then may have to adjust other items to fit around them, so don't pick a style based on the fonts and then try to add more header lines, for example.

Customization hint 2: The headers are pre-built with job variables that may not match what you use. Replace them with your own, or simply type text in their place, being careful not to delete or add pipe | characters.

## ***Multi-page document overrides***

In order to get the multi-page document to conform precisely to expectations, it is almost always necessary to override some of the document's full-sized settings, such as margins, spacing, etc.

In previous versions, the only way to do this was to use a multi-page settings file. This overrode all of the document's settings, and had to be constructed by taking your original document settings and making modifications to them, a process which required creating a separate user, experimenting with a document to achieve the desired results, copying that document's settings to the new user.ini, exporting it as a .set file, then going back to the original user.ini file and selecting the newly created multi-page settings file. This was extremely flexible, but awkward.

This version now has every document setting that users are likely to want to override built into the multi-page option dialog so that they can be overridden easily. Now, if you simply want to lines in the multi-page to be spaced out a bit more, just go to the multi-page dialog and select "line spacing" and make it slightly larger than the line spacing in the document.

The .set file mechanism is still there for users who need to override something that is not on the list, though that should be extraordinarily rare.

Most of the settings are numerical and are presented as a droplist/number box mechanism. The index columns and index threshold have been consolidated into this list, as well, to save space.

Here is a complete list of the new settings in the multi-page dialog and what they do. Note that any setting that is specifically an override of a document setting should be set to -1 if you want to keep using the original document setting for that metric. Metrics that did not exist in previous versions, even in the document settings, are marked with **\*\*NEW\*\*** and an explanation:

- **Page Format** -- overrides the document's page format since many multi-page formats prefer "page X" on the multi-page document but just "X" on the full-sized version.
- **Maximum text width** -- **\*\*NEW\*\*** performs an extremely valuable service: Wrangling unruly fonts into behaving. The text is no longer allowed to exceed this distance. If you have a large number of capital letters on a line, it no longer pokes the line out the right side of the box. Instead, it will shrink the font just for that one line so that it fits within this measurement.
- **Row 1 offset, Row 2 offset, Column 1 offset, Column 2 offset** -- **\*\*NEW\*\*** the distance that the rows or columns of miniaturized pages move up, down, left or right (can be positive or negative.) Use to fit the page more precisely inside whatever borders you have selected. Again, these numbers are set for you when you use a pre-defined format, but you can now tweak just these values without affecting anything else.
- **Top margin** -- overrides the top margin of the document. Note that this will NOT affect the top margin of the multi-page document. It affects the top margin of the individual

miniaturized pages. Extra space at the top of a multi-page output is generally not desirable, but can be accomplished by adding additional empty header lines (examples provided in the preset samples.)

- **Left margin** -- overrides the left margin of the document. In multi-page, this will be the binding margin on the left side before the left edge of the box.
- **Right margin** -- **\*\*NEW\*\*** Documents don't use a right margin since each paragraph uses its own margins. In order to dictate how much space there is between the right side of the paper and the automatically drawn multi-page boxes, it was necessary to add this metric.
- **Footer offset** -- **\*\*NEW\*\*** The location of the footer is determined by the page size and number of footer lines. The automatic position may not be precisely what you're looking for, so this number (which defaults to zero) gives you a way to tell the software to put the footer higher or lower on the page. Keep in mind that this will stretch the automatic boxes, so adjust this value after deciding how many footer lines you want, and before you adjust the row 1 and 2 offsets.
- **Header margin, footer margin** -- overrides the header/footer margins of the document. Does not affect the multi-page headers and footers, only the reduced-size individual page versions, as in Style 3.
- **Line height** -- overrides the line height of the document. Use this to stretch out the lines farther apart or put them close together.
- **Character spacing** -- overrides the character spacing of the document. Note that this only applies to fixed-space fonts, but that means that this can be particularly useful if you want to maximize the font size for the fixed-space portions of the transcript, since the spacing can be adjusted independently from the font height.
- **Line number margin** -- overrides the line number margin of the document.
- **Page number row, page number column** -- overrides the page number position of the document. Use in conjunction with the new page format field to customize what multi-page page numbers look like.
- **Timecode margin** -- overrides the timecode margin of the document. None of the built-in samples have timecodes appearing on the multi-page output. Timecodes are difficult to put on a multi-page output and still be reasonably readable, so they should usually be limited to the full-sized output. It is possible to squeeze timecodes in, but the fonts will have to be reduced and margins adjusted.
- **Paragraph number margin** -- overrides the paragraph number margin of the document.
- **Text box left, top, width, height** -- **\*\*NEW\*\*** Using the original text box from the full-sized document only works if you don't change any of the margins. Once you start changing the margins, line height, etc., it no longer fits. These adjustments allow you to specify an exact location for the multi-page version of the text box that is used to outline each separate miniaturized page, as in Style 3.

## ***Multi-page header/footer now support job variables***

You shouldn't have to type the witness name, the case number, the current date, etc., in three or four different places.

Not to mention the fact that the multi-page header/footer settings are permanent, and if you want them to be different for each printout, you have to manually go in and change them each time.

Forget all that. You can now put job variables from the form field system into the header and footer by using {VARNAME} anywhere in the header/footer.

So, for example, if you have a field with a variable called WITNAME where you type the witness, and another where you fill in the DATE (or maybe MONTH, DAY and YEAR) you can just put {WITNAME} and {DATE} (or {MONTH}, {DAY} {YEAR}) in the multi-page header/footer box.

The default Style 1-3 presets already hint at this by putting {WITNESSNAME} and {DATE}, etc., in the header/footer that are created when you select a preset.

## ***Multi-page full-sized cover page added***

As an added flourish for multi-page, a full-sized cover page option has been added.

Note that this is not a feature which prints the regular title page in full size. Instead, it's a new feature which prints a decorative cover page as the first sheet in a multi-page printout.

You can access this from the User settings/Programming tab, under "Cover page." If you leave it blank, no cover page will be printed. The cover page is always printed using an enlarged version of the default multi-page font, for consistency's sake. It also always prints each line centered, and centers the entire body of the text vertically. So, for example, you could type the following as your cover page...

**In the matter of**

---

**Joe Consumer, Plaintiff**

**v.**

**Very Large Corporation, Defendant**

**Case no. 123456**

---

**Deposition of**

**Millicent Bystander**

---

**January 16, 2015**

**at the offices of**

**Superlative Court Reporting**

**800-555-1234**

...and you would get a decorative cover page before your multi-page output.

Of course, what you REALLY want to type into this cover page is the following...

**In the matter of**

---

**{PLAINTIFF1}, Plaintiff**

**v.**

**{DEFENDANT1}, Defendant**

**Case no. {CASENUM}**

---

**Deposition of**

**{WITNESSNAME}**

---

**{DATE}**

**at the offices of**

**Superlative Court Reporting**

**800-555-1234**

...because this feature also supports using job variables.

Keep in mind that if there is a large amount of variation between the cover pages for the types of transcripts that you do, a template like this might not be flexible enough.

In this case, you can add additional variables to replace things like the words "Plaintiff" and "Defendant" just as you would on a title page.

In the worst-case scenario, if your cover pages are so variable that they cannot be templated, you can still make them document specific by simply putting a series of lines in the Cover Page that read {COVER1}, {COVER2}, {COVER3}, etc., for as many lines as you need, then add a series of comment lines in your document above your title page containing those fields. Whatever you fill into those fields would appear on the cover page. Worst-case.

## ***PDF output added***

And here we come to one of the primary reasons all that work was done on the multi-page output. Because we really want users to start using this feature and not to have to pay for Min-U-Script or e-Transcript or whatever.

It is no longer necessary to use PDFcreator or CutePDF or any other printer-driver-based system for creating PDF files.

Instead, there is now an "Output to PDF" menu item on the Production menu, right under the output to ASCII function. There is also a handy toolbar button which you might want to add to your toolbar. We'll certainly be adding it to the defaults, but only new customers and customers who import the default display settings will get it, so existing customers should go in and add it under the Window/customize toolbars function. (Add the "Delivery" button while you're at it!)

This brings up a "Create PDF" dialog which contains a number of PDF-specific options which will be enumerated below.

It also contains a "Print image options" button which allows you to access the details of how the PDF will be created. The reason for this is that a PDF is, fundamentally, a file-ized version of a printout, and you should be able to access exactly the same options for creating a PDF that you do for printing, including the draft watermark, the multi-page options, etc.

Pressing the print image options will reveal exactly the same dialog that you are used to seeing for printing, and these options all work exactly the same way. Now on to the new PDF-specific options.

## ***PDF digital signatures***

In the Create PDF dialog, there is a "signature file" item, followed by a "Browse" button. You can use this to select your digital signature file, which must be a **.pfx** file. For testing purposes, I created a self-signed signature by using the Adobe reader, but for the purpose of identification and data integrity validation, self-signed signature certificates aren't worth the paper they aren't printed on. You must purchase a digital signature from a TTP (Trusted Third-Party) validation service such as GlobalSign or EnTrust. Contact tech support for details.

Below that line, there is a "signer location" where you can fill in the location where the PDF document was created (signed.)

There is also a "signer e-mail" where you can fill in your e-mail address, which will be published as contact information in the signed PDF. You can also add other contact information here, if you wish, such as a phone number and/or address. Just be aware that PDF readers may not allow a huge amount of room to display this information in their dialogs.

## ***PDF concordance indexing***

If you create a PDF of a full-sized document with index or a multi-page document with index, the PDF file will contain the index as a separate pane of bookmarks with automatic access to each word in the document.

Opening the letter group shows each word starting with that letter. Clicking on a word triggers the query function on that word and will show a clickable example of every instance of that word in context in the document.

Opening the PDF will automatically cause the bookmarks pane to be displayed as a helpful prompt for the user that this functionality is available.

## ***PDF auto-indexing***

If you use the automatic indexing feature in Eclipse, there are now a couple of helpful hooks that will allow that information to be used by the PDF creation feature.

In order to use the auto-index material, first the PDF feature needs to know where the index items belong. Since that information is no longer present in the text when the index is inserted, the auto-index feature will now put in "hints" to the PDF feature telling it where the items came from. Starting in this version, when you generate an index, you will see index lines actually inserted into the index itself, at the beginning of each group, such as In:EXAM, In:PLEXBT and In:DFEXBT.

The PDF feature can detect when it comes across an index line and will insert it into the bookmarks under the appropriate index name.

For each item, it will show the page/line number where it appears, and have a clickable bookmark allowing the user to jump to that page.

Additionally, you can get the PDF indexing to create a bookmark for every mention of an index item, rather than simply linking to the specific items enumerated in the index.

If you add an index line immediately below the one naming the index itself, you can specify a wildcard, using the \* in place of the item. For example, you might have something like the following:

```
In:EXHIBITS
```

```
In:Exhibit *
```

```
INDEX OF EXHIBITS
```

Ex. No.	description	Page
1	A contract	6
2	A letter	14

In an index with this format, the index items are simply "1" and "2", or "A" and "B", etc., so searching for those isn't particularly helpful. The wildcard that was added, "Exhibit \*", tells the PDF creator to look for "Exhibit 1" and "Exhibit 2" and "Exhibit A", etc., as it's creating the document. Note that if your index items are already uniquely searchable (let's say your items actually look like "Exhibit 1" or "Exhibit 2") then your wildcard should just be \* and nothing else. The item is enough.

When it sees matching text anywhere in the document, it will make an additional bookmark for that instance, filed under the appropriate index type.

One final note about this feature: Index descriptions such as PLEXAM or DFEXBT aren't all that friendly for your clients to read. You can supply a friendly description for the index in a way that's similar to creating the header file, PLEXAM.ecl, that is used to label the actual text. Since it's a much smaller piece of text (it should be one line, abbreviated as much as possible) it needs its own file.

Create a file called PLEXAM\_DESC.ecl in your block files folder, and type into it something like "Plaintiff's Examinations" and that will be used as the label for the PDF bookmarks in that group.

### ***PDF embedded exhibit files***

No, we're not done yet. You can now embed exhibit files directly in your PDF transcript.

Go to any point in the document and use Insert print command (Alt+N) and you will see a new print command called "Attachment."

Select that command and hit [enter] and you will be given a file dialog. Select any file at all and it will be attached to the document in the current position. That's all you have to do.

It works best with other .pdf files, and many places insist on using PDFs for exhibits since the digital signatures allow the authenticity to be verified. However, you can just as easily embed .jpg pictures, .txt ASCII files, audio files, video files, etc. Whatever you like.

You can review that the attachment works by placing your cursor in the attachment print command and hitting [enter]. That should open the file just as a user would see it.

When you create the PDF, these attached files will appear in a separate attachments pane (the paperclip icon in Adobe Reader) and can be accessed directly that way.

You will also get a "Contents" bookmark group above the index which will list each of the files. Each file will have sub-bookmarks: One to open the file, and another clickable link describing the page and line number in the document where it appears that can jump right to it.

On the page where the attachment appears, a clickable hotspot rectangle will appear to the left of the line number where the attachment was added in the document. This hotspot will also open the attachment. So, between the attachments pane, the "open file" bookmark, and the clickable hotspot, clients using the PDF really ought to be able to figure out how to open the attachments.

### ***PDF embedded exhibit / auto-index linking***

Admittedly, the attachment documentation above is intended as the bare minimum you need to do to attach files to a PDF. Insert print command, Attachment, select file, done.

But it doesn't actually associate these files with specific exhibit numbers, or with particular index items, or indicate where they're mentioned in the document, or even put the hotspot rectangle around the appropriate text.

Surely, we can do better than that, right?

Well, yes, but you have to use the Eclipse automatic indexing features. Since some users don't, I didn't want users to think they couldn't do attachments at all unless you use auto-indexing. You can still do attachments, they're just not as cool.

In order to really make attachments shine, don't attach them in the body of the text; attach them to the index items after they're generated.

Let's revisit that exhibit index from the earlier example, but this time with attachments added:

```
In:EXHIBITS
```

```
In:Exhibit *
```

```
INDEX OF EXHIBITS
```

```
Ex. No.    description      Page
```

```
At:{DOC}Exhibits\Jones\12-13 Contract.pdf
```

```
1          A contract        6
```

```
At:{DOC]Exhibits\Jones\IMG2023.jpg
```

```
2          A letter           14
```

This same index now has an attachment added immediately above each index.

The PDF creator now knows what index items these attachments are associated with, and will automatically put the hotspot rectangle around the item, so in this case, there will be a rectangle around the "1" and the "2" (again, if you have verbose item names, like "Exhibit 1", the rectangle will be around the entire item.)

In addition, as the PDF creator is going through the document finding the index wildcards, it will not only make a bookmark, but will also draw a clickable hotspot around every instance in the transcript that will open the attachment, as well. (Note: The clickable hotspots don't appear on the multi-page output. There's just no space on a compressed page. The attachment pane is still available, however.)

## ***PDF encryption and protection***

The PDF format has a built-in mechanism for protecting the contents of PDF files. You can specifically allow or prohibit the user of the PDF file from printing, copying or annotating the file.

The Create PDF dialog has checkboxes for each of these options, allowing you to control what your clients are permitted to do with the PDF file once they get it.

At least, that's the theory. The reality is a bit harsher.

This works about as well as any other DRM (Digital Rights Management.) Entertainment corporations with billions of dollars in resources can't prevent 12-year-olds from stealing

music and movies. You're sure not going to prevent an attorney from printing a PDF transcript. It takes about two minutes to Google and download a PDF unlocker.

So, you can uncheck these boxes, but at best you're making these functions slightly more inconvenient. Then again, it is necessary to break the law in order to get around the restrictions, which should mean something to an attorney.

### ***PDF/A-1b compliance***

Please do NOT check this box unless you are required to. It disables many of the other features in the PDF export, such as the encryption that prevents users from accessing certain features, the search query bookmarks and the file attachment hotspots.

The reason for this is that PDF/A is a stripped-down version of PDF for the sake of compatibility and archiving. Court systems may ask for PDFs to be created in this format as future-proofing to ensure that the PDF files will always be readable in future versions of PDF readers.

Eclipse adapts to some of this by creating individual generic bookmarks for each index item, and the attachments are still available in the attachments pane, but other items are disabled.

### ***Transcript delivery dialog to integrate print/PDF/ASCII/etc. into one step***

At the bottom of the Production menu, under the Print/ASCII functions, is a new menu item titled "Delivery."

It's a huge, expansive feature, but it was designed to be as intuitive as possible, so my first suggestion would be to ignore the rest of this description and see if you can figure it out without instructions, which you probably can. Then come back here and read this and see if you missed anything.

If you have a document open, it will default to delivering that document. If you have no document open, you will be required to select at least one.

The Delivery dialog has a list of jobs to deliver (you can hit "Add a job" to add more) and a list of delivery tasks below it. Note that if you wish, you can select page ranges and volumes for each separate job that you're delivering by changing the page/volume text boxes.

When you first use the feature, a series of default delivery tasks will be created for you: Printing a full-sized document, printing a multi-page document, creating a full-sized PDF, creating a multi-page PDF, creating a page-image ASCII, Bundling the created files together, and e-mailing the bundle to clients.

By default, all of these tasks will be selected, and hitting the "Deliver" button will perform all of those tasks on each of the selected jobs.

If there are certain tasks that you want to skip, you can simply uncheck the boxes for the tasks you don't want to do this time before hitting the "Deliver" button.

Now, what if you want to customize? No problem.

Press the "Edit tasks" button. This will make the task list editable. The "Edit tasks" button will disappear and be replaced by a "Save changes permanently" checkbox. If you do not check this box, then any changes you make will be temporary changes that will only affect this one delivery. Otherwise, they'll be saved permanently in your user settings.

There are several ways to edit the task list:

Click the "X" button to delete the task from the list.

Click the Settings gear button to customize a task.

Click the "Add task" item to add a new task to the list.

When adding a task, you will be asked to select a task type from among the following: Print, ASCII, e-mail, Copy and Move. Each one has several additional options.

If you select "Print" you will be asked if you want a full-sized printout, Multi-page, index, etc.

If you select "PDF" you get the same options as printing.

If you select "ASCII" you will be asked what type: Page image, summation, etc.

If you select "Bundle" it will get added to the list with no further options.

If you add a bundle task to your task list, it changes the way that the system handles all of the files you're creating.

You might be creating an ASCII and a full-sized PDF and a multi-page PDF, and you'll get three separate files. If you have "e-mail" as one of your options, depending on the checkboxes you have checked, it will create three separate e-mails for these files. Even if you attached them all to one e-mail, the client would have to deal with three separate files.

With "Bundle" on your task list, once it's done creating all of the files (whatever they are) it will then create a PDF bundle containing all of those files.

This file will be named Jobname\_Bundle.pdf, and it will be a one-page PDF (it uses the Cover Page, adding some instructions for opening files) with bookmarks on the left for listing and opening each of the different files that were created.

If you have selected e-mail as a task, it will e-mail only the Jobname\_Bundle.PDF, with everything wrapped up in one convenient package.

For the Print/PDF/ASCII options, they will be added to the list after those simple selections. The default behavior will be to perform those tasks exactly as if you had issued the Print, PDF or ASCII commands normally. In this case, if you change something such as your standard ASCII setup or your multi-page print setup, it will apply to the delivery items as well. If you customize the delivery items further by pressing the task's settings button, the task will then have its own settings independent of the normal print/ASCII settings. That means that you can have the delivery function automatically print to two different printers, or print one transcript with timecodes and one without, etc. It also means that changing your normal print/PDF/Multi-page settings will no longer affect the delivery feature tasks that have been customized.

When adding or customizing the e-mail, copy and move functions, you will get essentially the same dialog for all three. It consists of a list of checkboxes for file extensions, such as TXT, PDF, ECL, NOT, etc., which you can select. It also contains a list of destinations. For the e-mail function, adding a destination will ask you for an e-mail address. For the copy and move functions, adding a destination will ask you for a file location. It will default to the file locations already set up, but you can scroll down to the "browse" item if you wish to use a folder that does not have a location defined for it under User settings/Programming/File locations.

There are no Copy or Move functions defined by default because those will vary tremendously depending on the user's preferences.

One suggestion for using these would be if you intend to deliver copies of ASCII files on removable media such as USB drives, in which case you could set up the Copy function to copy the ASCII and/or PDF files to the appropriate drive.

Another would be to use the Move function to move the job files (ecl, not, dix, wav) into a permanent archive folder after the delivery functions are complete. Note that if you're going to do this, don't use the BACKUP folder; that's intended for the automatic rotating backups and should be cleared out periodically. You may want to create an archive folder, assign an ARCHIVE=etc. File location to it, then direct the Move task to the ARCHIVE folder.

Some final notes on this feature: It will only deliver "jobs," not arbitrary files scattered to the four winds. In other words, it allows you to select transcripts (ecl files) from your JOB folder and nowhere else. All of the delivery features assume that the ASCII files, PDF files, etc., are all stored in the JOB folder for the purpose of e-mailing, copying and moving.

It automates the order of the tasks so you don't need to worry about that. Printing will come first, followed by ASCII, followed by e-mail and copy, followed by move. You will notice this if you add a print task; it will be added with the other print tasks. It's done this way so that tasks like printing to a PDF file and creating an ASCII is done before it tries to e-mail or copy the resulting PDF or TXT files.

### ***Continuous statistics updating***

Eclipse now continuously counts pages, words, lines, etc. There is a background process that's always running and is gradually moving through the document counting things. Every time an edit is made, it starts over. What will tend to happen is that during brief moments of inactivity, the statistics will be completely updated.

This has several implications.

First, it means that the total page count / last page number that appears on the status bar will be updated after a few seconds every time you make a change in the program. Previously, if you added two pages in the middle of the file, the status bar wouldn't change until you actually moved the cursor all the way to the bottom of the job. Now, it will update, showing a total page count two pages larger

Secondly, it means that the file statistics will update and be correct when you look at the file in the file manager, so information like the number of pages and the word count, etc., will stay accurate without having to explicitly execute the "job statistics" function.

### ***variable for note file creation date/time in form fields***

Rather than add a variable, the existing insert date/time function in form fields now uses the note file date rather than the current date. It will also use the timecode from the current location in the transcript rather than the current computer time. Note that in realtime, these will be the current date/time.

### ***EXAUTO for numbers or letters auto-incrementing***

In the same way that form field variables ending in NUM can auto-increment digits and ALPHA can auto-increment letters, using AUTO at the end of a form field variable can auto-increment either one, so a variable such as EXAUTO or PLFAUTO would count 1, 2, 3 or A, B, C depending on whatever was last typed in.

### ***Multi-page should be able to index speaker names***

Speaker names will now appear on the concordance index.

### ***Add ASCII import to dictionary conversion format list***

Previously, importing an ASCII into a dictionary had to be done through file/import. Now it can also be done through the conversion wizard.

### ***ASCII import should deal with timecodes***

Importing an ASCII file will now look for timecodes and will import them if found.

### ***ASCII import should have options for ignoring numbered header lines***

When importing an ASCII file you will get a dialog of options. The “import only the numbered lines” option has been changed to “import only the numbered lines, starting with” and then followed by a number. If the incoming ASCII file has numbered header lines, then you want to ignore those and start with line 2, for example.

### ***Allow student version to create RTF with almost nothing in it***

Some schools wanted to be able to teach students how to export and exchange files between CAT software. In order to protect the student version from being usable as an edit station, the RTF export was enabled, but only for the first ten pages of the file.

### ***Allow users to modify the index font size***

Under Print / Multi-page options, there are options for many different fonts and sizes. It is now possible to change the font and the size used for the index.

## ***Auto-reconciliation for ALPHASORT vs. NUMSORT***

Some indexes may have numeric or alphabetic entries, as evidenced by the need for the EXNUM, EXALPHA and EXAUTO form field variables. It is now possible to simply use the word SORT on an index line rather than ALPHASORT or NUMSORT. Just plain SORT will sort according to whatever the contents are. If it contains any digits at all, it will be sorted by the numeric value. Otherwise, it will be sorted in alphabetical order.

## ***Some way to export the notes from an .ecl file***

When in a translated document file, you can now go to File/Export and among the file types is .not (notes) so that you can directly export all of the notes from an Eclipse file to create a new translatable note file. This was previously only possible by juggling a bit by exporting an RTF and importing the RTF into a note file. The direct export is more convenient.

## ***Send e-mail option in Create ASCII dialog***

Under Production / Output to ASCII, there is now an "e-mail" checkbox. If you check that box when creating the ASCII, then after creating the ASCII it will open up an e-mail and attach the file to it.

## ***Allow non-student users to turn on "School mode" to try extra features***

The student version of the software has all features enabled, including voice input, Bridge output, captioning, etc. It has built-in limitations on translation, printing and output in order to make the software usable for educational purposes, but not for professional purposes.

If an owner of the professional version of Eclipse wishes to try out a feature they don't have access to, such as a steno reporter wanting to try out voice, or a scopist with an edit station wanting to experiment with steno translation, it is now possible to temporarily put the software into student mode in order to be able to access all features in a limited way for the purpose of experimentation.

Note that it is recommended that you create a separate experimental user.ini file for this purpose since the documents opened in this mode cannot be used in the professional version.

In order to activate student mode, go to Help/About and press the "School mode" button. This will enable school mode for the rest of the execution session. Note that in order to go back to the professional version, you must exit the software and re-enter it.

## ***Allow students to output to Bridge Mobile without a license upgrade***

Students need to be able to try all of the Eclipse features without having to purchase upgrades. The student version prevents professional use by limiting pages, time and adding watermarks and educational version messages to all methods of outputting data.

## **Auto-Brief**

### ***Auto-brief max words setting***

By default, the auto-brief feature will scan for phrases up to five words long containing words not on the ineligible words list. Some users felt that even five words is too many for a brief and wanted to dial that down.

You can now specify the maximum number of words that the auto-brief feature will include in a suggestion.

### ***Allow AB background window to change color***

Changing the background color of auto-briefs in the color selections will now change the background color of the auto-brief window.

### ***Word-wrap briefs in the pop-up window to a maximum width***

Under User settings / Translate / Auto-brief settings, there is now a “max width” setting. By default, it's set to 0, which disables the feature. Setting it to a particular value forces the auto-briefs to word-wrap within that limit.

This is particularly helpful when using the auto-brief window on a small second screen, which has become popular. If you know that your second screen is 600 pixels wide, for example, then set the auto-brief width to 600 and the briefs will never extend past the width of that screen.

### ***If user asks for another brief use alternate forcing theory***

This makes the +STENO syntax of the auto-brief rules considerably more useful. You can now have multiple +STENO lines, such as:

+OEUG

+OEUB

+OEUZ

If you do this, the first suggestion for any auto-brief word or phrase will add the steno from the first + line, so you might write “stenographically” in five strokes and get SOEUG as a suggestion.

If you use {ABNEW} to ask for a new suggestion for that word, it will rotate to the next line, suggesting SOEUB instead. Hitting {ABNEW} again rotates to the next line, etc.

## ***Auto-brief settings dialog should have buttons to link to related programming tab items***

Under User settings / Realtime / Auto-brief settings, there are now buttons reading “Ineligible words” and “Theory settings” which will directly open up the editor window for modifying those items normally under User settings / Programming (Auto-brief ineligible words, Auto-brief steno theory.)

## ***Option for number of times word/phrase must be written to produce a brief***

This is probably a bad idea, but it was easy to implement, and people asked. Normally, the auto-brief function will suggest a brief if it sees a word or phrase written twice. Some users wanted to increase the threshold. Under User settings / Realtime / Auto-brief settings, the “Suggest briefs automatically” option has now been changed to “Suggest briefs automatically after X occurrences” where the X defaults to 2 and can be increased to larger numbers. Keep in mind that if you change this to 4, for example, that means that you will have to write a word or phrase the long way 4 times before the auto-brief feature will suggest a shorter way to write it (unless you specifically request a brief.)

## ***Right click context menu on auto-briefs: reject/new/open***

If you right-click on an auto-brief, you will now be given the option of rejecting that specific auto-brief or asking for a new brief (these are equivalent to the {ABNEW} and {ABREJECT} / {ABDELETE} dictionary entries.) You also have the option to open the auto-brief dictionary.

## ***Auto-brief plural detection should reverse spelling adjustments and work for requests***

If you request an auto-brief for a plural word, it will now give you a brief for the singular under the assumption that you can easily add a plural suffix to it if you wish.

Also, plural words will be transformed backwards to singular using spelling rules such as those in the prefix/suffix rules, so unpluralizing "testimonies" will produce "testimony."

## ***ABNEW should put the new suggestion on the same list it deleted it from***

In other words, if the system offers a suggestion and you use {ABNEW}, the new suggestion will still appear on the “suggested” briefs list. If you request a brief for a word or phrase and then hit {ABNEW} because you don't like the steno it gave you, it will put the new item on the “requested” briefs list.

## ***{ABDELETE} feature to remove a brief defined as the next stroke written***

To use this feature, a new metadictionary entry is required:

```
{ABDELETE}={/?ABR}
```

Writing {ABDELETE} while in realtime will immediately display the steno machine emulator on

the screen, but this is a special situation: It will only look at the very next stroke you write and then will immediately resume normal realtime. The stroke you write will then be removed from the auto-brief dictionary and the auto-brief window. The {ABREMOVE} stroke only removes the last brief on the list, but with {ABDELETE}, it is now possible to remove any brief on the list just by hitting {ABDELETE} followed by the brief you want removed.

### ***{ABSELECT} now allows accepting a specific suggested brief***

This works exactly as the new {ABDELETE} function. It requires a new metadictionary:

```
{ABSELECT}={/?ABS}
```

If you have User settings / Realtime / Auto-brief settings / “require accept stroke” turned on, then suggested briefs are not immediately applied to translation and must be accepted first. {ABACCEPT} only accepts the last brief on the list, but if you want to accept a brief higher up on the list, you can now hit {ABACCEPT} followed by the stroke for the brief that you wish to use. It will then be moved to the “requested” list and will be available for use immediately.

### ***{ABCHANGE} requests new steno for a specific brief***

Much like the {ABSELECT} and {ABDELETE} items, this one allows asking for new steno for a specific auto-brief, rather than {ABNEW} which always acts on the most recent suggestion.

Place the following in your dictionary to accomplish this: {ABCHANGE}

That stroke will prompt you for the steno stroke representing the auto-brief that you want a new steno suggestion for.

In order for this to work, you must have the following item in your metadictionary. This will be part of the default metadictionary in the .set file distributed with the release version:

```
{ABCHANGE}={/?ABC}
```

## Translation

### ***Add Passport Touch writer menu item defaulting to 115K baud***

There is currently no other difference between the Passport and Passport Touch input selections other than the 115K baud rate, but it does provide a differentiation mechanism for the possibility of future features that will only work with one model and not the other.

### ***Passport shadows option***

If a user doesn't want to see shadows on the note file, the chances are that they also don't want to see the extra shadow data file in their list of data files. This change addresses both issues.

In the "read notes" dialog, if you are reading notes from a Passport, you now have the option to read in the shadows or not. Since they are only currently used for display purposes, there's no harm in not reading them.

### ***Button in Translation Magic settings to link to programming tab items***

Under User settings / Translate / Translation Magic, there are now two more buttons: Exceptions and Phonetics. These will directly open up the editor window for modifying those items normally under User settings / Programming (Translation Magic Rules and Phonetics table.)

### ***Continuous redacted text mode for translation***

Some users have requested a way to write in redacted text for a long period, such as a bench conference. It is already possible to disable realtime output, but there was no way to obscure what appeared on your own screen. Depending on your display colors, you can make it so that redacted text is not visible on even on your own screen.

One thing to be careful of is that redacted text is a text type, just like translated text, typed-in text, untranslates, conflicts and form fields. If you tell the software to treat everything you write as redacted text, all of these other types will be overwritten while that mode is on since a piece of text cannot have multiple types. Conflicts in that mode cannot be selected, etc.

The redacted text mode, like many other modes in the software (such as literal case) has dictionary entries for ON, OFF and TOGGLE in case you want to use one entry to turn it both on and off (not recommended.)

The suggested dictionary entries are {REDACTON} {REDACTOFF} and {REDACTTOGGLE} and use the following metadictionary entries:

```
{REDACTON}={/?RDO}
```

```
{REDACTOFF}={/?RDF}
```

{REDACTTOGGLE}={/!RDT}

These metadictionary entries will be made part of the default .set file in the release version.

### ***Translation Magic rule for 'contains steno'***

In the Programming/Translation Magic Rules, you could previously enter a line such as

-FPLT

which meant to never replace the -FPLT stroke, no matter what, even if there was a really good misstroke or phonetic match for a longer entry. You can now enter items such as

&RBGS

which means never to replace a stroke that has RBGS in it at all. Use this carefully. After all, what if you tried to write SPHEURBGS and you wrote it SPWHEURBGS instead? The above rule would mean that Translation Magic wouldn't have permission to fix it.

### ***Spell mode abort strokes should be user-definable***

What this refers to is the fact that when using spell mode or stitch mode, every stroke is interpreted phonetically, meaning that you can fingerspell something like “widget” with two strokes (WEUD TKPWET) instead of six. Normally, you have to hit the spell mode or stitch mode toggle stroke to turn them off. In these modes, as a backup in case you misstroke or forget to toggle it off, hitting certain extremely common strokes is used by the software as an indication that you are done spelling, such as -RBGS, -FPLT, STKPWHR- or -FRPBLGTS. These are not only common, but are extremely unlikely to be used as part of a fingerspelling process. These strokes previously were not user-definable.

In this version, rather than add a whole new feature just for that very subtle safeguard, it will look for any stroke that appears on the Translation Magic rules list. In other words, any stroke that should never be touched by Translation Magic also should probably not be touched by the spell mode or stitch mode automation features, either, and should always be treated as their plain dictionary definitions.

### ***Insert missing BY line needs to differentiate between Q and A by lines***

There has been an “answer by lines” feature in the software for a while, and an “insert missing by line” feature, but they didn't work well together at all. This version can supply missing answer by lines as well as missing question by lines, and will correctly supply the appropriate speaker separate in each case.

### ***A way to have close paren not trigger a paragraph punctuation***

The TPC three-letter metadictionary code, which has existed for a long time, will now prevent automatic paragraph punctuation in all circumstances regardless of the punctuation mark actually being added.

This means that if you wish to create a specialized dictionary that prevents automatic

punctuation from being added, it is possible to do so by using the three-letter code. For example, the following entry:

```
{) }={/<"/)?TPC}
```

would add a closing paren, but treat it as a terminal punctuation, so hitting {Q} or {A} or {N} after it would NOT cause a period or question mark to be added after the close paren.

This is required for some situations in which the paragraph ends in a narrative, such as (nods head) which some users prefer not to have a period either inside or outside the parens if it's the last thing in the paragraph.

### ***A locked conflict should not display as unselected when chosen by “pick 1st choice.”***

A locked conflict is often one where the first choice is assumed except under specific circumstances in which it should pick the second choice. For example, an `\exhibit\Exhibit{#N}` conflict that is programmed only to pick the second choice if followed by a number. In these situations, the pick first choice possibility is to be expected, so the conflict should not appear in the “unselected” color that would normally indicate that the A/I couldn't find an appropriate choice.

### ***Conflict AI-sharing***

Often, users will have many different stylistic conflicts that should follow exactly the same grammatical rules.

For example, if you have conflicts for `\page\Page{#N}` and `\volume\Vol.{#R}` and `\exhibit\Exhibit{#N}`, those are all pretty much the same in terms of grammar rules.

The same goes for conflicts such as `\yes{,?}yes{,?}` and `\no{,?}no{,?}` and any others that may have different content but identical grammatical contexts.

For these items, it is now possible to link those conflicts together to share AI data. The way this is done is to pick a single "master" conflict, such as KWE for "yes" that is going to act as the repository for all similar conflicts.

That way, the "no" and "of course" and "naturally" and all of the other words that you might want to put into a `\word{,?}word{,?}` structure like "yes" has, can follow the same rules.

To assign those other conflicts to use the same rules as the "yes" conflict, define them like so:

```
TPHO = {=KWRE}\no{,?}no{,?}
```

In other words, use the slop stroke syntax, but add on the correct text to use. The "slop stroke" feature will then know to share the AI data, but use different text.

What will happen is that the translator will use the same AI data for each of these conflicts, and when you are editing and you pick any of these conflicts, the new AI data created by your selection will all go into a central place.

The end result will be that these conflicts will all learn much faster and become more accurate more quickly, and be more consistent, because they are not each learning separate rules and are instead sharing a single set of rules.

### ***Make Stenograph pedal equivalent to + key***

Older Stenograph writers had an optional pedal that could be held down while a stroke was pressed. Previously, it would be converted to a steno mark. However, since what the writer is actually doing is adding the flag to the stroke, it makes more sense to make it act like one of the “extra” keys on extended keyboards like the Passport, Gemini, Treal, etc.

### ***Add primary language selector to settings***

Under User settings / Edit / Spell options there is a new setting called “Primary transcription language” which offers English, French, Italian, Spanish and German options (each in their native language.) This option controls all of the built-in language-specific processing features, such as the special translation grammar translation rules. Additional languages can be added as necessary when new language translation rules are added. Note that selecting a primary language is not necessary for languages for which there are no language-specific rules.

### ***Autopunctuation shouldn't try to change the punctuation unless the sentence is relatively simple***

The auto-punctuation feature that attempts to detect whether the current sentence is a question or a statement and add a question mark or period appropriately based on the grammar of the sentence is a bit hit-or-miss. This version dials back the feature, so even if it's turned on, it won't attempt to make a change unless the sentence is only a couple of lines long.

### ***Delayed number triggers***

By default, number triggers will only appear to the immediately adjacent number. However, some patterns of speech may indicate an upcoming number of a particular type that may not be adjacent. For example, if a question asks “what is your date of birth” that implies strongly that the answer will contain a date. However, it may appear after other text, such as “I was born on 121598.”

It is possible to convert that to a date by writing {#D} immediately before a number, but that's an extra stroke. It is possible to add the {#D} date trigger to the word “date” making it “date{#D}” instead. However, since the number isn't immediately after it, it won't apply to the date in the answer.

In order to get the trigger to apply to a future number, you can now use {#D>} in the dictionary entry. That will simply apply the date trigger to the next number that is written in the translation.

## ***Date formats should allow for an optional automatic trailing comma***

The {#DY} and {#DO} date triggers, which are often attached to the month names, will now convert the number and will add a soft comma afterwards automatically, so if the next entry is a normal word, it will be followed by a comma, but if the next entry is a terminal punctuation or paragraph, the soft comma will be ignored just as any other entry defined with {,?} at the end of it.

## ***Option to put integral prefix/suffix in untran color***

There is now a new option under User settings / Translate / “Show integral pre/suf as untran”

With this option on, if a word is translated by virtue of the integral prefix/suffix feature, it will be shown in untranslate color to call attention to the fact that it was translated without a perfect match in the dictionary.

## ***Button near integral pre/suf translation setting to open programming tab***

Under User settings / Translate, next to “Integral pre/suf” there is a “set” button that will directly open up the editor window for the integral prefix/suffix settings normally under User settings / Programming.

# Realtime

## ***Auto-block feature for boilerplate text***

If you work in any situation where speakers of any kind will occasionally pepper their remarks with the use of prepared texts, this feature is for you.

It's intended primarily for reporters who work in court where judges may use dozens of prepared statements such as jury charges, etc.

The way it works is that Eclipse can now detect that you are writing something that matches one of these prepared texts and will allow you to pick one off the list so that Eclipse can complete it for you.

For example, you might write "As the jury in this" and the software will show a pop-up window containing numbered lines "1: (Criminal Jury Charge.ecl) ...case, you will be the judges of the facts..." "2: (Civil Jury Charge.ecl) ...case, you will be..." etc.

You can then pick one, and the rest of the text will be filled in. Note that you can write as much as you want before this happens, or simply continue writing. Eclipse will always pick up from where you stopped writing the text.

Here's how it works:

Create a folder for these prepared texts. Create "descriptive file name.ecl" files in that folder.

Under User settings/Programming/File locations, add an AUTOBLOCK line such as

AUTOBLOCK={JOB}Blocks/Auto or AUTOBLOCK={JOB}Prepared texts or  
AUTOBLOCK={MAIN}Shared auto-blocks or whatever.

It should NOT be the same as your regular BLOCK folder because you don't want it auto-completing whole title pages, certificates, appearances, etc., just based on the first few words happening to match what you're writing.

Put the following line in your Metadictionary (this will become part of the defaults for new users)

```
{AUTOBLOCK*}={/"%/?AUB}
```

Add lines to your main dictionary for selecting auto-blocks. You probably only need a few numbers, since the list will be narrowed down depending on how many different blocks match the text that you have written so far:

A\*UB 1 = {AUTOBLOCK1}

A\*UB 2 = {AUTOBLOCK2}

A\*UB 3 = {AUTOBLOCK3}

etc. (the actual steno is entirely up to you.)

Once you have made these preparations, any time you are writing realtime and you see the auto-block pop-up window, you can hit one of the {AUTOBLOCKx} dictionary entries with the number of the block you want to use, and that will be filled in automatically.

### ***Deny saving should be permanent setting***

In addition to the “deny saving” button that can be pressed while in realtime, there is now a “deny saving” checkbox option in the realtime output dialog that can be checked and left on permanently. This will periodically send a deny saving command to the receiving computers while they're connected. It does this periodically because on a serial connection, an attorney who hooks up late might not have received the “deny” command and would therefore be permitted to save the file.

### ***Bridge filename should transmit periodically***

Bridge Mobile doesn't use this feature, but Bridge does, since it can connect through a serial port which doesn't imply a file name. A user hooking up late would miss the initial transmission of the filename, so Eclipse will now re-transmit the file name every page so that when the auto-save kicks in it uses an appropriate file name.

### ***Bridge refresh ON should be the default***

The Bridge output type will now send edits unless told otherwise.

### ***Add a setting for using the Windows default audio setting instead of using a specific device***

Selecting the User settings / realtime / compression option will now show a (default) item as the first selection. This selection specifically tells the operating system to use whatever device is selected in Windows as the default recording device.

### ***Send script line dictionary entry that doesn't require a macro***

This was created because the keyboard macro realtime output sends all macro keystrokes to the receiving program, meaning that realtime macros can't use macros that are supposed to execute Eclipse functions. However, that means that the keyboard macro output system had no way to support scripting.

A new three-letter code has been added to the metadictionary system. To use this feature, add the following line to the metadictionary:

```
{SCRIPT}={/?CCS}
```

Then create a dictionary entry containing simply {SCRIPT} to the dictionary. Hitting that dictionary entry will send the current script line to the output system. This will work both for the keyboard macro output and other output types, as well.

### ***Make "start from which stroke" realtime message optional***

In the translation dialog, there is now a "Prompt for starting stroke" option. If you turn this off, it will always start from the last stroke in the writer. Note that this option is permanently stored and will then apply to the use of the "Instant realtime" button, as well.

### ***Division interval should be able to expire when no writing is occurring***

When using the division interval, it would previously only expire while writing, but now it can expire during a pause and write out a division at the appropriate time even if the user isn't writing at that exact moment.

### ***Output COM port recovery***

If a COM port is unplugged or disappears (which can happen when a USB-to-serial adapter is unplugged) Eclipse will now periodically attempt to close and re-open the port in order to try to re-establish the connection.

### ***Username/password and reconnection automation for TCP-IP connections such as Speche***

Some direct TCP-IP connection systems such as Speche require typing a username and a password. Previously, this required using the Direct Output feature to manually enter a username[enter]password[enter] macro sequence, which was a bit awkward.

This version prompts for an optional username and password when setting up the TCP-IP address. In addition to automating the username/password sending when connecting, if the connection is dropped, it will re-transmit the username and password when reconnecting.

### ***Add date preset drop-down list under the Realtime filename template field***

The realtime file name can use any of the time/date template commands, and to help remind users of that, several additional sample templates have been added to a droplist.

Also on the droplist is a help item that will immediately bring up the help page with the list of all of the templates and what they mean.

Note that in order for the "auto-rename realtime jobs" feature to work, the filename must start with RT and the third character must be a digit, which is why the templates are all set up with RT immediately followed by one of the day/month/year items in digit format. Keep that in mind when designing your own.

## **Dictionaries**

### ***Analyze documents dictionary report of text that appears with different steno***

In addition to the dictionary conflict report that shows all of the ways in which one piece of steno is defined as different text in different dictionaries, there is now an additional report.

The second portion of the ANALYZE.ecl file will now be a report of all of the words and phrases (or other dictionary entry text items) that are separately defined in the different dictionaries, and the different steno that is used to write them.

The word will appear with each separate steno variation appearing on a separate line.

This allows you to easily compare multiple different ways that you have defined text.

### ***Add phrases to analysis.txt using auto-brief rules***

In the ANALYSIS.TXT file created by the analyze documents feature, after the suggested capital words, followed by the suggested lower-case words, there will be a short space, followed by a list of capitalized phrases, then a list of lower-case phrases.

These phrases are collected based on the auto-brief rules, and represent phrases that you do not currently have dictionary entries for in the dictionaries that you selected for the analysis.

Note that these phrases will often contain words that you do have entries for, so that you could write them as phrases. It is intended as a way to call your attention to phrases that happened multiple times in a document that you might want to consider making a brief for ahead of time for a future translation.

### ***Multi-select on analyze documents feature***

When analyzing documents, it is now possible to multi-select dictionaries and documents to analyze by using the Ctrl+click, Shift+click, etc.

### ***Ignore variations in analyze documents***

The analyze documents function now disregards two categories of words that are likely simply variations of existing words: words ending in 's, and words in all caps that are already in the list in another case. This means that "Greene," "Greene's," and "GREENE" will no longer be treated as three different words.

### ***Don't apply suffixes in analyze documents feature if not in English***

The function that automatically applies suffixes to analyzed document words will be skipped if the currently selected language is something other than English.

### ***Dictionary merge should have 'keep all' option***

When block reading one dictionary into another, when it detects a conflict between an incoming entry and an entry already in the current dictionary, there is now an additional option. The options now are “Overwrite / Make conflict / Keep original / Stop asking.” If you select “Stop asking” you will be asked what to do with ALL of the remaining entries that conflict, and will be asked “Overwrite / keep original,” and the answer will be used for all remaining items with matching steno. Note that there is no option to create conflicts with all of the remaining entries that use the same steno, since that is extremely unlikely to be desirable.

### ***Dictionary print should keep steno/text column order to match display***

Some users re-order the steno and text columns so that the text column appears first on the screen. This display change and others will now be reflected on the dictionary print output.

### ***Dictionary print should use dictionary name as document name***

The document name isn't relevant for most printing, but when creating a PDF it appears on the PDF file itself, and previously appeared as a generic name for all dictionaries.

### ***Dictionary print should allow variable font size***

The dictionary printout will now be proportional to the dictionary display font size, just as the notes printout is.

### ***Dictionary spell check should look for substrings***

Previously, the spell check would only work on entries containing a single word. This version will check each word in phrase dictionary entries.

## Editing

### ***Find speaker function should work for by lines***

The find speaker function previously only worked specifically on “Speaker” paragraphs, but will now work on by line paragraph types as well.

### ***Find feature should optionally wrap around to top of document and continue to starting point***

When the end of the document is reached, you will get a message stating “end of document reached. Stop searching? (yes/no.)” so that you can stop there by answering “yes”, or answer “no” to continue searching though the whole document. Checking the “don't show this message again” box will prevent this message from appearing and will always stop at the end of the document exactly as previous versions did.

### ***Find feature should optionally remember last search string***

The previous search string will appear in the find dialog, but highlighted so that typing a new search string can be done with zero extra keystrokes simply by starting to type.

### ***Find feature remembers anywhere/word/etc. if selected using radio buttons***

If you type asterisk wildcards, such as \*ing to find a word ending in “ing” it will change the radio button automatically. When this happens, the radio button continues to be automated. However, if you select the radio buttons using the mouse, the dialog will remember which one you selected and will maintain that selection for future searches until it is changed manually (or until the asterisk is used again.)

### ***Allow user to hit "Shift+Ctrl+," in the globaling window to get {,?}***

This is an augmentation of the feature that allows you to hit Ctrl+, to get {,}

### ***Multi-layer jump to last edit point***

You can now hit the jump to last edit point function repeatedly and it will return to your previous edit point, then the previous edit point before that, then the point before that. It keeps up to five long distance markers for times when you have hit movement commands that jump the cursor a long distance.

### ***'Selective' global prompting for each instance like a search and replace***

There is now a “Prompt” checkbox in the globaling dialog. If you check that before accepting the global, then it will display each instance of the replacement in the document and will ask yes/no if you want to replace it for each instance.

## ***Indexing wildcards should treat lockspaces and spaces the same way***

These items are equivalent in terms of the use of the content and only exist for changing the formatting, so the indexing should treat them the same way.

## ***Permanent bookmarks***

The text markers created with the Move / Advanced / Markers function are now permanently saved in the document and can be used in future editing sessions.

## ***Add blank field dialog options to add font/attributes after creating field***

Any blank field can be boldface, underlined or have a different font by virtue of block marking it and changing the font or attribute after first creating the field.

Many users expect there to be some way to change those things in the "add blank field" command, so this version streamlines that process while still foregrounding the concept behind the way it's actually working.

There is now a "Change font" checkbox in the "add blank" function. If you check that, then once you're finished with the other options and you hit OK to create the field, it will automatically be highlighted and the font change dialog will appear, allowing you to select any font/attribute you like for the field without extra steps.

## ***Add a way to apply a macro to a list of files***

This was originally requested as a specific feature, but when I looked at it I realized that it could be accomplished with a small variation to the macro system, while also allowing many additional functions to be accomplished using this powerful addition.

In the macro list dialog, there is now an "apply to" button. This will only work if you have selected one and only one macro from the list. It will display a file dialog and allow you to select as many files as you like.

Once you hit OK, the software will open the first document in the list and immediately run the macro. Once the document closes, the software will open the next document and run the macro again.

If you make sure that your macro ends with Ctrl+Q or Ctrl+F4 to close the document, it will cycle through every one of the documents you selected without you having to do anything at all. I mention this as an option in case you would prefer to review the document to make sure the macro applied correctly before closing the document yourself, at which point it will still automatically go to the next one.

The original feature that was requested was a way to apply a particular set of modified document settings to a large group of block files. This now makes it possible. Export the document settings to a newdoc.set file, for example, then write a macro that does, approximately, User settings/User/Import/newdoc.set/OK/OK/OK/Ctrl+Q. Select that macro (it doesn't even need a keystroke or dictionary entry) and use "Apply to..", select all of the block

files you want to change, and hit OK and they'll all change.

I'll look forward to hearing any other innovative uses for this feature that users come up with.

### ***Optionally allow Auto-globaling misstrokes to use slop stroke syntax instead***

This is an interesting one. Imagine that you have the Global Magic feature turned on (and why wouldn't you?) and that you perform a global on the stroke STHAP, which should have been “snap” but you missed the initial P. It will probably display “snap” among the global magic suggestions. You could simply select that, but what if you wanted to define STHAP as {=STPHAP} instead? This would indicate that that dictionary entry is actually a slop stroke for “snap” and not a good entry for “snap.”

Slop strokes are especially helpful for things like speakers, which may change, or conflicts, since slop strokes use the original entry's A/I, rather than having no A/I at all.

Now, when the Global Magic suggestions appear, you can hit F9 (think of it as the “view dictionary” keystroke) to toggle the suggestions window. It will then show the slop strokes (and the text they would translate) instead of the text. Selecting that numbered suggestion will insert the slop stroke syntax into the global window for creating the final dictionary entry.

### ***Partial auto-replacement***

Some common typing errors are not whole words like “hte” which should be “the” but are instead parts of words.

You can now enter prefixes and suffixes on the auto-replacement list by using the \* symbol as a wildcard. For example the following auto-replacement:

```
*toin=tion
```

indicates that if you type “toin” at the end of any word, it will be changed to “tion” instead.

### ***Undo function for join paragraphs should restore paragraph attributes in the process***

Previous versions used to split the paragraphs back up again but would make them fixed lines or generic paragraphs. This version restores the original paragraph type, so undoing a join function is much more accurate.

### ***Reading a file into another should create paragraph types that don't exist***

Originally, when reading in a block file such as a title page, if a paragraph type existed in the block file that did not exist in the current document, that paragraph wouldn't work at all and would be turned into a Fixed line.

This version will add missing paragraph types into the current document so that block/read will always work correctly. Keep in mind, however, that if the paragraph type does already

exist in the current document, and the margins for that paragraph are different in the file or the current document, the current document's settings for that paragraph will be used, not the settings in the file being read in.

### ***Allow copy current/master to do labels and fonts based on paragraph remappings***

Copying current to master normally ignores fonts and paragraph labels entirely, since every speaker name in a job is a paragraph label, and every single font change, even to just a few characters, gets added to the document. Most of these will not be used in future documents, so copying them to the master format makes no sense.

On the other hand, this version has been changed so that if a paragraph label or a font is used as the default font or the default label for a paragraph on the list, then that label or font is copied over to the master settings since the paragraph setting is a strong indication that that font or label will be used in future documents.

### ***Toggling open and close double/single/double quotes***

Hitting the single quote (apostrophe) character while editing will toggle open/close in the same way that the double quote character works.

### ***Alt P Q D should work with one-space punctuation if declared in metadictionary***

If the metadictionary entry for {.} says to follow the period with one space (as is appropriate when exporting to other software for publishing in a proportional font) then the Alt+P function that changes the terminal punctuation at the end of the current paragraph to a period will ensure that that period is followed by a one space instead of two.

### ***Hyphen in editing should not add a lockspace if removed in metadictionary***

Some users prefer not to use the lockspace before the double dash, so if the lockspace is not in the metadictionary, it won't appear during editing, either.

### ***Backspace should remove all punctuation declared in metadictionary***

In a piece of text such as “He. Went home” you can place the cursor on the “W” and hit the backspace key to remove the period, the extra space and down-case the “W.” The punctuation marks that are eligible for this sort of automation used to be hard-coded within the software. This version now looks at all of the punctuation definitions in the metadictionary, meaning that any punctuation mark that has a metadictionary entry associated with it can be removed automatically with the backspace-from-a-distance function just like the period.

### ***Allow directional quotes in editing if declared in metadictionary***

If the metadictionary has the open-quote/close-quote dictionary entries { “ } and { “ } and are defined using the “smart quotes” which are correctly directional, that will affect the editing as well. Pressing the “ key at the beginning or end of a word, or block marking a word and hitting the quote key will then use the appropriate smart quotes in the text.

### ***Down-cap in editing if there is no cap command for metadictionary***

If the metadictionary entry for a particular punctuation mark doesn't have a cap command in it, then when that punctuation is typed before a capped word or is used to replace a terminal punctuation mark such as a period, the word will be down-capped if the lower-case version is correctly spelled.

### ***Wrappable non-space characters like - and /***

This is a hidden feature which should mostly be left alone, but it now exists for the purpose of experimentation, and to allow one small tweak.

The user.ini settings file now contains a line reading “WordWrapChars=etc.” and contains a list of characters that represent acceptable places to break (word-wrap) lines in a paragraph.

Previous versions would word-wrap only on spaces. This version now adds the tab character as an appropriate word-wrap point. Some users have indicated that they would like for the software to be able to word-wrap on hyphens or slashes. This can be done by adding those characters to the list, which can currently only be done by editing the .ini file directly.

If it ends up being a popular feature, it can be added to the user interface. Keep in mind that this could theoretically cause previous documents to print differently if there were places where it would not have word-wrapped before but don't word-wrap now.

## **User interface**

### ***Extended edit controls in dialogs***

Most users know some of the hotkeys that can be used in some dialog edit boxes, such as Ctrl+W to get a special character, or Ctrl+K to change capitalization.

Over the years, more of these handy hotkeys have been added, but there's no visual indication on the screen that they can be used, no way to look them up other than the help text or the manual, and no indication of which edit boxes actually allow these extra functions.

All of that has now been addressed with a single enhancement: All extended edit boxes will now appear with a "<<" button placed at the right side of the edit box. Press this button to see a context menu showing all of the possible enhanced editing functions. You can select items from this menu, but you can also use it to look up the keystrokes and familiarize yourself with how to use these functions without the menu.

As a bonus feature, a "Google it" menu item has been added to this menu which will perform a Google search for the contents of the edit box, or the marked block inside if, if applicable.

### ***Build index item should remember previous entries in drop boxes***

When using Edit / Insert / Index item, the index name and the location format are now remembered so that the next time you use the dialog you don't have to type those entries if re-using the same ones over and over.

### ***Keystroke dialog droplist should contain number keypad combinations***

The number keypad delete key is different from the regular delete key, but still deletes the keystroke from the hotkey window, so the droplist is necessary to define it.

### ***Job report should be a non-modal dialog box***

The Tools / Job report function dialog that shows the list of comment lines can now be set aside, you can click on the text and edit it, then go back to working with the job report list. This is much more convenient for dealing with things like proofreader comments where further adjustments to the text might be necessary without having to re-run the job report data collection process every time.

### ***Job report allows sorting of the commands by text or chronological***

The job report dialog now has a "sort" button. By default, the list appears in chronological order. Hitting "sort" changes them to alphabetical order.

### ***Allow super/sub definition to specify percentage of main font size***

When block marking text and selecting Format / Text attributes, the superscript/subscript

options, in addition to having an option for vertical offset to tell the text how much to move, now also contains a percentage option that will determine what percentage of the current font size will be used for superscripts or subscripts. For example, a superscript set to 75 / -100 will be 75% of the size of normal text, and will move up 100 twips higher than normal text.

### ***Use steno dialog in slop strokes table***

Under User settings / Programming / Slop strokes, you can now use the “Add” or “Modify” buttons to easily modify the entries. The steno dialog will appear, first with the “before” stroke, and when you hit OK, with the “after” stroke, making it easier to enter slop strokes into this table more accurately.

### ***Alphabetize programming tab items***

So many items have been added to the programming tab over the years that it makes more sense to alphabetize them than to leave them in the arbitrary order they were in before.

### ***Right-clicking on a toolbar should take user to customize toolbars***

It will automatically bring up the customize dialog with appropriate toolbar selected.

### ***Edit box should support reverse / and ?***

Under user settings/Edit, there's a “reverse / and ?” option which allows typing a “?” without hitting the shift key. That now works in edit boxes within dialogs, such as during globaling.

### ***Special characters dialog should contain a font button***

You can now hit the “font” button in the special characters dialog and select the font you wish to see the special character for. This is especially helpful if you wish to see if something like the Symbol font has the special character you need.

Note that this does NOT cause the font in the document to change to that font type. This is because the special character dialog can be used in the document and in dialogs, and each different usage for this dialog may require a different mechanism for changing fonts, or may not support fonts at all. For example, when globaling, or in a dictionary it might be necessary to use {F:Symbol,13}D{n} in addition to just typing the D to get the delta symbol. In a document, it's necessary to block mark the character, select Format/Font and select Symbol.

### ***Internet browse menu item function should handle URL encoding***

Whatever gets typed into the internet feature automatically gets URL encoded. What that means is that certain characters that are illegal in URLs are automatically replaced with their coded equivalents. For example, if you attempt to search for “R&D department” it would normally be rejected because the & character is a special command that should be followed by a variable name. This version will replace it with %26, which is a special command meaning the & character.

# Spell check

## ***Permanent watchwords list***

Under User settings / Programming / Spelling exceptions, it is now possible to add words that are added to the watchwords list for each job (they will appear as WATCH=word1,word2,word3,etc. in the job variables.) To add these words to the spelling exceptions to have permanent watchwords, use the ! symbol before them:

!accept

!except

Watchwords, as a reminder, have two features: First, the spell check will always stop on them. Second, once you have run the spell check even briefly, at least once, the watchwords will appear with the warning underline on the editing screen just as misspelled words are.

## ***List errors should be a non-modal dialog***

You can now move the list errors dialog to the side, click on the text and do some editing, and then go back to the list errors dialog and continue reviewing the list of errors. This could be a significant improvement since it would not be necessary to re-run the entire list errors function and re-generate the list if it was necessary to make an editing change in the middle of the review process.

## ***List errors should use marked block only, if applicable***

If you block mark a section of text and select the Production / List errors function, it will only list the errors in that block.

## ***Add Google lookup button to spell check dialog***

When spell checking a word, you can use the "Definition" button to show the dictionary definition supplied by Eclipse. Inside the definition dictionary, there is a "Google it" button that can be used to immediately do an Internet search on that word.

## ***Add Google suggestions to Global Magic and spell check lists***

The Google servers have a way to retrieve just the autocomplete suggestions for any piece of text. Since this is usually quite good at guessing what you're trying to type even if you mistype it badly, this is something we can use.

If Eclipse detects an active server connection, and if you have Global Magic turned on, then each time you go to global it will quickly check with Google as a last resort, and will add a Google autocomplete suggestion as the last item on the list.

The highest priority items are still the regular Global Magic items calculated by looking at the spelling dictionary and possible mistrokes for your own dictionary.

The biggest use for this is proper nouns. Aren't sure how to spell Gujarati or Marrakesh or Schuykill? Take a stab at by writing it phonetically, and when you go to global it Eclipse will send the phonetics to Google and see what it comes up with.

The spelling error dialog has room for many more than ten suggestions. After all of the spell dictionary suggestions, the spelling suggestions dialog will now list " from Google:" and several additional lines with whatever Google thinks might fit the misspelled word you typed in.

### ***Spell check-as-you-type for dialog edit boxes***

If you have the User settings/Edit/Spell options/Display errors while editing option turned on, you will now get an extra feature: Certain edit boxes will now show spelling errors.

This isn't relevant for all edit boxes, so it's only been enabled for some of them, such as the globaling and find dialogs. You'll see a red underline, just as with editing. Note that cursor and mouse movements will make the error indicator go away in order to make it a bit less disruptive.

If you wish to get spelling alternates for any of the words indicated as misspelled, you can place the cursor anywhere in the word and use Ctrl+F for "Fix spelling" (Ctrl+S is already being used for suffixes!) This will give you the spelling alternates and allow you to fix the word.

So, now, let's say you're filling in a form field on your appearance page and you need to type an address. You take a wild stab at the spelling of "Shukill" and it comes up underlined, indicating a misspelling. You hit Ctrl+F, scan down past the spell dictionary items to the Google suggestions, which has "Schuykill" among them. That's the one. Pick it and hit Change and continue filling in the form.

## Speakers

### ***By lines should use colloquy colors***

If speaker names in the speaker table are given individual colors, those colors will now apply to the by lines containing those speakers.

### ***Ability to change speaker names in blocks***

You can now mark a block of text and then use F2 (Speaker) to display the speaker table. If you use the “Change” button to change one speaker to another, there is now an additional option. Besides “All” or “forward only” you can now select “Block” and it will only change the speakers within the marked block.

### ***Speaker table changes should trigger refreshes***

When a speaker name is changed from SPEAKER01 to MR. SMITH, or from MR. JONES to MR. GREEN, that change may only exist in the table of paragraph labels, so the text itself isn't actually changing. However, since what appears on the screen is clearly different, it has a bearing on the Bridge output screens. Now, when making a speaker change, if it's during realtime and there's a Bridge output with Apply edits turned on, each speaker paragraph using the changed speaker will be re-transmitted to the Bridge output.

### ***Speaker table entries in dictionaries should work with “apply globals”***

When you use the speaker table and enter in master/current speaker equivalents, such as SPK01 = MR. SMITH, that is stored in the job dictionary (if you have the job dict “save in” checkbox turned on in the speaker table dialog.) This is not a typical dictionary entry, but is instead an entry specifically for populating the speaker table. Tools / Apply globals will now apply these entries to the speaker table and the document.

### ***Form field / speaker table relationships added***

**Speakers to Fields:** When starting a translation, many users enter in their standard speaker designations. For example, the speaker table might have SPK01 and SPK02 as master speaker names to go with the dictionary entries such as STPHAO = {S:SPK01}. If you use those exact same speaker table master speaker names as form field variables, that data will be filled in automatically. If enter MR. SMITH for SPK01 in the speaker table, then a form field with SPK01 as the form field variable will automatically be filled in with MR. SMITH, as well.

**Fields to Speakers:** In order to create the association between speaker names and form fields, use a technique very similar to using the NUM, ALPHA or AUTO suffixes. The SPK suffix indicates that the current form field is a speaker. If a variable end with SPK and you have “prompt for contents” turned on, when whenever you fill in the field, whether manually or from a list file, it will open the speaker table with the contents of the field already typed in. To assign it to a speaker, use the cursor down key to the appropriate row and hit [enter].

# Audio

## ***Audio level button functionality for newer versions of Windows***

The record level and playback level buttons have been disabled in the last couple of versions of Windows. This version now checks which Windows version you're using and substitutes newer versions of those functions.

There is no direct OS function to get to the record level, so the record level button will simply jump to the sound device properties, which saves the step of going to Control panel/sound. Going from there to the levels can be done by selecting Recording/Microphone/Properties/Levels.

## ***Multi-channel audio recording/playback***

Eclipse can now record and play back multiple separate channels of audio without using any special hardware or file formats.

Use case examples:

- 1) Multiple audio devices leading to separate, individual microphones to create a multi-channel recording wherein each channel is a separate recording from only that microphone, such as one for the room, one for the judge and one for the witness.
- 2) Multiple recordings being made from the same audio device simultaneously. For example, recording a low-quality, small-file format for the purpose of streaming to a remote scopist, but recording a high-quality, large-file format for the reporter to use for their own editing that will not be sent over the Internet.
- 3) A separate voice audio backup for voicewriters, allowing channel 1 to be the room audio, and channel 2 to be a WAV copy of the same mask or microphone voice audio that Dragon Naturally Speaking is using for the voicewriter recognition.

It does this by using the same audio infrastructure it used in previous versions: WAV files recorded from built-in and/or USB audio devices. This version simply allows you to have multiple recordings going on at the same time.

A multi-channel audio recording will have a separate WAV file for each channel. When only using one channel (as in previous versions) it will be jobname.wav. Further channels will be named jobname\_2.wav, jobname\_3.wav, etc. For now, the system is limited to 8 separate channels, which really should be sufficient. Keep in mind that each one could be recording in stereo for a total of 16 separate audio channels.

The audio user interface has undergone significant changes in order to allow this new functionality.

Under User settings/Realtime, all of the audio options have been moved into a new dialog that is accessed by pressing the new "Audio recording.." button.

This dialog contains the following controls:

The Auto-pause, Auto-restart and Levels buttons work exactly as before.

The Audio Device drop-list allows you to select what device the current channel is using to record.

The Select Compression button allows you to select what codec to use for the current channel.

Note that both of the above controls work exactly as previous versions did if you are using only one channel of recording.

However, you can now hit the "Add channel" button to add new channels to the list. New channels will default to PCM, 11khz. The list will show the channel number, the device being recorded, the codec, and the storage rate in megabytes per hour (bytes per second seems less useful.)

When you select a channel on the list, you can then use the device selection drop list and the compression button to choose the device and codec for that channel.

The "Delete channel" button will delete the currently selected channel. Note that you cannot delete all of the channels.

Note that it is possible to make multiple recordings from the same device source, as described in the use cases. If you want to record high-quality/low-quality, create two channels with the same device input but different compression selections. Voicewriters should use channel 1 for the room audio and can select the voicewriting USB device (such as the Andrea) for channel 2, even though Dragon is using it at the same time.

If you wish to use multiple microphones in separate locations in the room, keep in mind that this sort of multi-channel recording requires a separate audio device for each microphone. For example, to record room, judge and witness in three separate channels, you could use your computer's built-in microphone in channel 1, then use a USB audio device with a microphone plugged into it for the judge, and a separate USB audio device with a microphone plugged into it for the witness. You can use a USB hub if your computer doesn't have enough USB ports. **IMPORTANT:** Multiple audio jacks are NOT separate audio devices. If your computer or your USB device has a microphone jack and a line-in jack, those are NOT two separate devices and you cannot use them to record two separate independent channels.

Also note that for all features in Eclipse other than record/playback, the software will ONLY look at channel 1. For example, the feature that copies segments of the .wav file to the DIVISIONS folder, the audio sharing through Connection Magic, the level indicators on the infobar and the realtime window, etc., are all features that will only reflect what is going on on channel 1. For that reason, channel 1 should always be the primary recording. That will be the copy that your scopist gets automatically. You can still give a scopist the extra channels manually at a later time, which would be recommended anyway since it will take significant additional time to send that many additional WAV files.

Under User settings/Edit, there is now an "Audio playback.." button that accesses the multimedia control panel.

This dialog contains the same speed/threshold controls that it did before, but new controls have been added:

The Channel drop-list allows you to select which audio channel you are playing. The default is 1. There is also an "all" selection which will play all the channels at once. You might think that using "all" should be the default, and of course you can set it that way, but for many applications, this would not be good, such as the simultaneous high/low quality recordings or the simultaneous room/voice recordings. And even for the multiple microphone setups, be aware that if there is any microphone cross-talk, playing all channels simultaneously could result in odd-sounding echoes depending on the performance characteristics of your hardware and software. Multi-microphone setups should consider the possibility of either a unified room mic as channel 1 or a mixed version of the sound as channel 1, leaving that as the default and only using the separated channels in order to disambiguate overtalking when the sound is otherwise undecipherable. If you intend to use the "all" option to play all of the channels simultaneously, it is strongly recommended that you make sure that all of the channels are using identical compression settings.

The Device drop-list allows you to select which audio device you are playing through. Most of the time, you should leave this at the default, which will play through your computer's speakers. However, some users plug in separate USB audio devices that have their own playback headset ports, and you may decide that you get better playback with that.

If you find yourself frequently switching playback channels in editing, you may find it advantageous to create some hotkey macros that perform [control panel/channel X/OK] all in one keystroke (check the "play on exit" option to eliminate the need to add the play button to the macro.) It is likely that macros like this will be added to the default macro list for convenience for the release.

In addition, there is an "auto" checkbox next to the channel selector. If you have that checked, then when you hit play anywhere in a document, Eclipse will look at the paragraph that the cursor is on. If it's an Answer, it will assume that the speaker is THE WITNESS (or whatever you have entered under User settings/Translate/Convert ANSWER to.) If you're on a colloquy, it will know the name. If you're on a question, it will scan back to the most recent by line. So now that it knows who was speaking, how does it know which audio channel to play? In order to know that, it will have to know where the speakers were in relation to the microphones that were set up around the room.

That's where the speaker seating chart comes in. Open the seating chart, and if you have auto-channel selected, the seating chart will ensure that you have a separate MICROPHONE object for each audio channel that you recorded. Drag the microphones to the approximate locations where the microphones were in the room relative to the speakers. Once that's set up, hitting the play button will always play the audio from the microphone closest to the person who was speaking at the cursor position.

## ***New multi-channel recording player interface***

Another option for multi-channel recording is The Liberty Court Player, a multi-channel system for recording audio in a courtroom, along with video. It can record each microphone in the room in a separate audio channel, allowing you to isolate individual voices during playback.

Details of the player are here:

[http://www.libertyrecording.com/LCR\\_main.htm](http://www.libertyrecording.com/LCR_main.htm)

You can download the player here:

<http://www.libertyrecording.com/download/lcr70pe.msi.zip>

Keep in mind that the player is free, but the recording software is a commercial product. If you are considering doing multi-channel recording, you should contact the company and find out more about the kind of equipment that will be required. Normally, they work with courts directly. If you already work in a courthouse that uses this system, you're all set. If not, you may want to discuss with them options for setting up a system for yourself, possibly with portable equipment, and what the cost might be for such a system, hardware and software included. Audio recording is an area with a great deal of variation depending on the customers' needs.

As far as Eclipse is concerned, the new feature in this version is that Eclipse will support synchronized playback using the Liberty Court Player. In other words, if you have a multi-channel recording recorded with this system, you can load the recording into the player in the background and Eclipse will recognize this. When you are editing the transcript in Eclipse and you hit the play button, instead of playing a .wav file, it will command the Liberty Court Player to play the multi-channel recording at the timecode corresponding to the cursor location.

Setting this up does require a couple of extra steps since the interface between the two programs is itself in the early beta test stages. If you are interested in trying this out, contact ASI at [support@eclipsecat.com](mailto:support@eclipsecat.com) and we'll get you the additional instructions that are necessary. They are not being added to this changes.txt file because they are likely to change as the final adjustments are made to the software.

# File management

## *New file dialog created from scratch*

In previous versions, anywhere Eclipse asked you to open an existing file or save a new file, it would present you with the standard Windows File Dialog. Unfortunately, this dialog has a lot of problems.

First of all, it has become increasingly buggy. Users may have noticed when creating a new file the dialog will ask up to three times "create file X (yes/no?)". Also, in Windows 8.1, clicking on a sub-folder will often cause a "create file X" message (even though you're not wanting to create a file at all) just because of the default file name. Also, in Windows Vista/7/8, this dialog now eats the first few dozen characters out of any macro that attempts to use them, so writing a macro to do Block read / filename / enter doesn't work. You have to put something like Block read / 30 backspaces / filename / enter. It's a bit awkward.

Not to mention, that dialog has many design problems. Over the years, it has become bloated and confusing. Do we really need forward and back buttons like an Internet Browser? The fact that the back and up buttons (both textless arrows) are entirely different things is confusing. Do we need a hierarchical folder display on both the left and the top? Does the one at the top really need a virtual/actual disambiguation that displays different results on it depending on whether you select from the arrows or click on the text? Does it need a refresh button? Do we need a preview pane button for a preview pane that doesn't do anything? Do we need a new folder button? Do we need to see every possible folder in the entire system? Do we really need to be able to delete, rename and move files, particularly when such things can be done accidentally when simply trying to open a document (click/drag has caused many users to accidentally drop a file or folder somewhere, and double-clicking too slow triggers a rename.)? Do we really need to see a "type" column when we told it that we want to open a dictionary? Of course they're all dictionaries! Do we really need a drop-list for the filename that always shows us all possible files with the current name, even though we've specifically told it we only want to open transcript files?

That dialog is, in technical terms, a hot mess.

That's why many pieces of software replace it with a dialog of their own design that is more carefully tuned to the needs of the users of that particular software product. So, that's what I've done.

One thing to be aware of: If you are using macros that operate these dialogs, many of them will no longer work. However, many of them are no longer necessary. For example, many users would change Alt+E to a macro and would change "Open text" to Ctrl+Alt+E. The Alt+E macro would do Cmd:Open text, then shift+tab twice to put the cursor in the file list instead of the filename box (why Microsoft assumes that we always want to type a name from scratch, even when specifically opening an existing file, is beyond me.) Macros like this are no longer necessary because the "open" functionality always defaults to the file list instead of the name box.

Here is how the new dialog works:

You will notice that, while dramatically simplified, it also retains much of the Windows visual design elements just for the sake of familiarity. No one should feel completely lost.

If you desperately need to get to the Windows version of the dialog for some reason (such as the desire to go to an unusual location, or a desire to actually use the explorer-like functions such as rename or move) you can simply hit the "Browse" button and it will jump into the original Windows dialog.

When performing any sort of "Open" function, the keyboard focus will be placed in the file list, meaning that you can immediately type letters or select with arrows to find the file you want, then hit [enter].

When performing any sort of "Write" function, such as block write or export, the cursor will be placed in the "Name" box, since you are likely to want to type a new name for the file in question.

Selecting a file from the list will fill in the name box. Typing a name in the name box will highlight the first file in the list that matches, or no file at all if no file matches. This allows you to easily cross-check for accidentally typing a new name that matches an existing file.

On the left side of the dialog, instead of a list of every possible location your computer knows about, including the entire folder hierarchy tree, the "Location" list now contains only those locations that the Eclipse software cares about, such as JOB and BLOCK. Selecting one of these items will immediately update the file list. Note that this will also show subfolders inside these locations if there are any. If you do use subfolders for organization, you should consider creating actual file locations. If you have subfolders such as Realtime or Archive in your job folder, you could use User settings/Programming/File locations and add `REALTIME={JOB}Realtime` and `ARCHIVE={JOB}Archive` and then those would appear as separate items on the list, complete with their own subfolders, if applicable.

The size of the folder pane and file list can be modified by dragging the border between them.

There is still a drop-list on the lower-right that can be used to select different file types. When opening a specific type, such as a transcript, there will be only one option. When doing functions such as an import, it may have several, just as before.

The file list itself will show the file name and extension, the file size, and the creation and modified dates. It will also show an icon for the file. This icon will usually match the icon that Windows explorer shows, but it will often show an Eclipse-specific icon that is more appropriate. For example, Eclipse doesn't register the .mac extension itself, but some systems see it as a "MacPaint image" even though Eclipse is using it as a portable macro file. Eclipse will show a Hyperkey icon for that file, rather than a photo or picture icon.

The name/size/created/modified columns can be clicked to sort the files forward or backward by the information in that column. The columns auto-resize to fit the content. Each folder you visit will store the sort column and forward/reverse order separately, so you can have your jobs sorted by creation date and your blocks sorted alphabetically, for example. Note that the folder headers will show "<" and ">" symbols to indicate which column is being sorted, and in what direction.

This is an Eclipse dialog, not a Windows dialog, and as such, it can take advantage of the

"customize" functionality for resizing, changing the zoom, anchoring the position, or adding transparency.

This dialog is now completely under my control, so any further refinement is now possible, which was not true of the Windows version. If you would like to see additional functionality (such as an info pane like the file manager) that is now possible, so feel free to pile on additional suggestions for making this better.

### ***File manager should have tabs for all folders specified in File Locations***

Must like the new file dialog, the file manager now lists all of the file locations. It puts the JOB folder first for convenience and lists all others alphabetically after that.

Note that the destination, similarly, has the built-in Prompt / e-mail / burn to CD / Browse.. options, but is then followed by all of the locations available under File Locations.

Keep in mind that this does give you tremendous flexibility for creating your own organizational structure. You can now specify arbitrary file locations entirely of your own design and they will be available anywhere that files need to be accessed.

### ***Add {DESKTOP} file locations (like {DOC} for documents)***

You can now use {DESKTOP} in the File Locations feature, making it possible to create folders such as EXPORT={DESKTOP}scopist\RTF files

The DESKTOP folder will also appear in the new file dialog and the file manager.

### ***'New folder' functionality for the file locations feature***

Under User settings / Programming / File locations, it is now possible to use the "Modify" or "Add" functions to easily manipulate file locations using a simple dialog rather than typing the whole path from scratch. Note that the "Add" function will add the location simply using FOLDER=etc. So you will have to replace that with whatever you're creating an entry for, such as WAVREC or DIVISION or whatever.

The folder dialog that appears has a "new folder" button so that it's possible to create the new folder ahead of time in addition to being able to select existing folders easily.

### ***Button for Programming/file locations in User settings/User/Advanced***

In User settings / User / Advanced there is now a button for "other file locations" that will directly open up the editor window for the file locations normally under User settings / Programming.

### ***Transcripts should remember creation software version number***

The creation version is now recorded separately from the version that last edited the file. This information is visible in the file manager.

## ***File manager should offer to close all appropriate file types***

When copying files in the Eclipse file manager, files must be closed since Windows cannot copy files while they're opened and being worked on. Previously, if a job file was open the file manager was ask if you wanted to automatically close it. It will now do this also for note files, dictionaries and wave files.

## ***Translate dialog should contextually decide to use "open" or "save" dialog***

When doing a tran/edit or a quick tran, and selecting a note file, it's more appropriate to be selecting an existing .not file. However, when doing realtime, it's more appropriate to be typing in a whole new filename, so the dialog will change subtly (putting the cursor in the filename box by default instead of the list.)

Likewise, when selecting the .ecl file, that will always assume saving a new file and will ask you to type a new filename rather than selecting an existing one.

## ***File locking feature to prevent future changes***

This is more removing a feature than adding one. Windows allows you to set a file to read-only using the file manager or Windows explorer. However, many users previously backed up files to CD-ROM, which is a read-only format. When these files were restored, they were left as read-only, which created a support call every time.

To eliminate all of these support calls, for a long time Eclipse simply removed the read-only flag whenever it attempted to open a file.

Backing up on optical media is no longer as popular as it used to be. This version will no longer attempt to remove the read-only flag. It will remain a read-only file. That means that it is now possible to deliberately protect a file from being overwritten by making it read-only.

Just to review, it is possible to access this from the Eclipse file manager. In the file manager, select the file, hit the "properties" button, then check or uncheck the "read-only" flag and hit OK.

## ***When working on a read-only document continue to remind users not to make edits***

Each time the software detects that the user is attempting to make a change to a read-only document, the warning will appear, rather than simply appearing when the document is first opened.

# Captioning

## ***SMI format conversion***

There is now a SAMI format conversion (.smi files) among the file formats available when in a job a under File/Export/file types. These are used for post-production captioning on Windows Media player files.

## ***Apply timecode offset to .scc file export***

The .scc files contain timecodes describing when the captions should be displayed. These timecodes now have the document's timecode offset applied.

## ***External COM port pass-through enhancements***

The external COM port feature will now interpret the data enough to create a record of what the user was writing in realtime.

It will show a readable sample in the captioning preview window, though not an exact copy of what appears on the viewer's screens. It only shows text and linefeeds, not positioning, indentation, colors, or other decoder-specific formatting.

It also takes this data and feeds it into Eclipse document files as follows:

After starting a realtime session, the user can optionally create a file named "extpreview.ecl" (A macro to create and open this file is recommended.)

If the translator can see that this file is open, then any data coming in from the external COM port will be added to this file so that the user can see what the captioner is writing.

If the external output is on, the data will ALSO be sent to the output AND the realtime file. Script lines will also be sent to the realtime file.

In summary, a caption coordinator should have three windows open: The script, the extpreview file, and the realtime file.

The script will show the material that can be transmitted by the coordinator at any time. The extpreview file will always show the external input in its entirety. The realtime will show only what has actually been transmitted from either source, depending on whether the user has hit commands to send script lines, or whether the external output was activated.

When switching from scripts to realtime, it's possible that the preview window mentioned above could have some text already written that should appear on the air.

It is now possible to use the "send script line" function to send all of the text in a marked block. This feature makes it possible to switch from script to realtime and simultaneously transmit some of the already-written text.

The ideal way to do this would be with a macro. Assume that the coordinator has left-clicked

or otherwise navigated to the spot in the extpreview file where the realtime should resume. The macro could block mark starting with that position (or perhaps move Home first to the beginning of the line) then hit the "end of document" command to block mark from there to the end. Use the Send script line to send that text, then the Edit toggles, enable external output, OK, then switch windows back to the script.

That macro would make it possible for a coordinator to click on the screen with the mouse and hit one button to basically resume realtime from a point in the text that was written a few seconds ago before the switch occurred.

### ***Script lines copied to text document should use current timecode***

When you run realtime scripts to the output system, such as with closed captioning, the script lines get added to the realtime file. The update now makes sure that those copied script lines have the current timecode so that the final file has correct and consecutive timecodes throughout.

### ***Add UI for CC block/pass settings***

For some encoders, in some very specific situations, it is necessary to manually change the initialization codes sent to the encoder during the process of putting it in block or pass mode.

These codes are a sequence of control characters and text, and are now accessible to users who want to modify them and know what they want to change them to.

To change the codes, go to the output dialog, and next to the "mode" label, before the droplist, is an "edit" button. If you have "block" selected, you will be editing the block command. If you have "passthrough" selected, you will be editing the pass command.

These commands look like this: `^C^A7 C1^M`, using the same syntax as the Tools/Realtime/Direct output command, in that `^A` means "Ctrl+A". Note that `^M` is equivalent to a carriage return.

This is obviously an extremely technical feature that users should mostly stay away from unless advised by a tech support person or a TV station engineer familiar with the idiosyncrasies of their particular hardware/setup combination.

## Speech

### ***When using speech TM should automatically be disabled***

Since Translation Magic does not apply to speech users, it is now automatically disabled.

### ***Disable number formatting***

This option under User settings / Input / Speech setup automatically disables the number formatting features in DNS entirely, meaning that Eclipse will get the raw number words instead of converted numbers. This eliminates the need to disable it in the DNS settings. This is one of those options that doesn't work in older versions of DNS.

### ***Use dictation mode***

This option under User settings / Input / Speech setup sets DNS to dictation mode automatically whenever a session is started, eliminating the need to set dictation mode manually each time. Note that this only works with the professional version of DNS.

### ***Use realtime timecodes***

When you turn on this option under User settings / Input / Speech setup, Eclipse will read the timecodes in realtime from the phrase hypothesis system, just as it did in previous versions. This does cause the slower drop rate, but hopefully creates timecodes that are more accurate and reliable for the playback and correction process. By comparison, turning this option off should create a faster drop rate, but the timecodes are a bit more wobbly, meaning that when the recognition rate is bad, the timecodes are often way off and may not allow for accurate playback and correction.

### ***Put in time adjustment for key/voice duplexing***

There is a new option in the User settings/Input/Speech/Setup for "Key/voice duplex offset." This setting is in milliseconds.

This is an enhancement to an existing feature that allows interleaving of voice and keyboarding. When doing voicewriting, the "force translation" feature will intentionally delay forcing the translation so that it enters the translation stream at the same time that the words word said, rather than forcing it into the stream immediately.

That way, it's possible to create one-key macros such as Q to force-tran {Q}. By creating macros to do paragraphing and punctuation, it could be theoretically possible to completely eliminate having to say qmack, amack, quex, cah, perk, spee-one, etc., for all of the paragraph and punctuation. Instead, a user could speak naturally and use keystrokes for everything not spoken.

This feature never quite took off because the timing was always a bit off. This new feature may make it practical by allowing users to adjust the offset forward or backward. If the

keyboarding is consistently appearing too soon, increase the value. If it's appearing too late, decrease it.

It was a relatively easy feature to add. I'll be interested in hearing if anyone experiments with this any further and if they have success.

### ***Speech should be able to start session with mic off***

Some users would prefer that speech sessions start with the microphone turned off, so there is now a "start with microphone on" option in the User settings/input/speech/setup dialog that can be turned off if you wish.

### ***SpeechText output should disallow content on selected paragraphs***

If you are a voicewriter and you export a raw text file while you have a SpeechText.dix file available, this is an indication to the system that you are creating a voisteno text file for analysis.

Since this file is solely for the purpose of running the analyze documents feature to teach the speech engine context, it would be ideal if this file did not contain non-spoken parts of the text such as form pages and prepared block files.

In order to accomplish this, this version will now ask which paragraph types you wish to include in the export, and will default to the testimony paragraphs, such as Q, A and Colloquy, and will omit Fixed lines and style paragraphs.

You can select or de-select any paragraphs to be included and these selections will be remembered for future exports.

### ***Presence of SpeechText.dix file should determine whether Raw Text export contains linefeeds***

Many speech users use the File/Export/Raw text feature to create processed speech text files for the analyze documents function.

When using the SpeechText dictionary, this allows re-processing of items like question paragraphs back into voisteno such as "qmak" so that the analyze documents function knows what voisteno command word would go in that location.

This process also strips out all carriage returns so that the Dragon Naturally Speaking program doesn't assume that the user is going to say "new paragraph" in that location.

However, some speech users are using the Raw Text export to actually export raw text for production purposes.

This version will look for the SpeechText dictionary, and if it's not there, will output the Raw Text exactly as a steno output would, with all linefeeds intact. If the SpeechText dictionary is there, it will perform the extra processing.