

Introducing Total Eclipse 5

Page	Topic
2	AutoMagic
5	Info Bar
7	Auto-Briefs
	Suggestions, Requests, Reminders - Separate color coding
	Font
	Optionally show auto-briefs as untrans
8	Minimum phonetic elements
	Acceptance stroke
9	Archive: Job-specific auto-brief dictionaries
	Phonetic display of auto-briefs
	Minimum stroke savings
	Reminders for commented entries only
10	Auto-brief can work for numbers and dates
11	Auto-brief should detect plurals and suggest singulars
11	File Management
	New text document file storage system. Relationship to older versions.
12	Maximum of 10 Auto-Backups for text, main dictionary, and user settings
	Job info expanded
13	Migration of user's files to "MyDocuments" folder
15	Editing
	Show spelling errors as you edit
	e-book proofreading
	Text globals now selected via checkbox in globaling dialogue
16	Form field math formulas for billing pages, etc.
17	"Find" dialogue can search for macro strokes and hidden characters
18	Doc Utility commands separated (Set Type, Quote, Unquote, Format Script)
19	Translation and Realtime Transcription
	Integral prefixes and suffixes applicable during translation
	Text replacement dictionary entries now apply to multiple entries
20	New number conversion format for comma within date
	Automatic continuation format for doubled paragraphs
21	Serial input passthru, Bridge text global support, CART window improvements
	Division interval
	Slop strokes table can work with partial matches
	Audio
22	Audio recording sent through StenoLink connection
23	Division intervals offer Incremental backup of WAV file
	"Sync audio here" button
24	Dictionary Management
25	User Interface
26	Writers
27	Indexing enhancements using regular expressions

Introducing Total Eclipse 5

Total Eclipse 5

New features and enhancements made since 4.3.0.10

AutoMagic

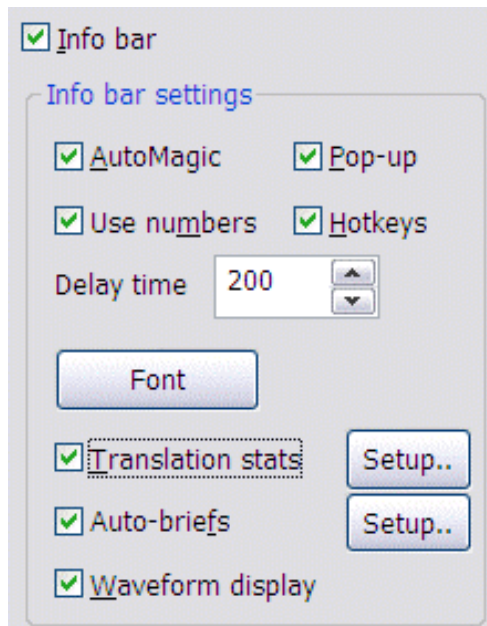
Finally, Eclipse can do your editing for you.

Well, not quite, but close. You still have to press one button for each edit.

The concept behind AutoMagic is simple. There are two main goals:

1. Make it possible for a new Eclipse user to be able to competently operate the software without having to learn anything.
2. Make it possible for experienced Eclipse users to streamline their editing by reducing keystrokes.

> **Setup in "View Toggles" Dialogue:** The feature can be turned on and off in the info bar settings under User settings/Display/View toggles. The info bar is usually turned on for AutoMagic to work.



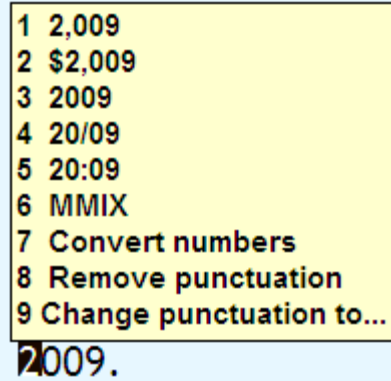
The basic idea is this:

No matter what the user is doing at any moment, the software will attempt to anticipate the most likely commands or options that the user might want in that location. It will display those options in the info bar labeled from 1-9, and the user will be able to select one of those options or execute one of those commands simply by pressing the number key on the keyboard. It could theoretically be possible for a user to become proficient at using Eclipse without knowing anything other than how to move the cursor and press number keys.

When the suggested operation is an actual command that also appears on the menus, the toolbar icon will appear next to the command, and the keystroke or Hyperkey for that command will appear on the right, just like a menu item, in order to encourage the user to eventually learn the keystrokes for these functions.

> **Delay:** As you move the cursor through the job, the AutoMagic items do not appear immediately because you might be moving the cursor rapidly through the document. It will wait until you stop moving the cursor for at least half a second (500 milliseconds) before it shows suggestions. Under the view toggles, you can set the number of milliseconds it takes between moving the cursor and having the AutoMagic choices appear. Once that time has elapsed, you'll see the suggestions.

Introducing Total Eclipse 5



> **Pop-Ups:** Under the View toggles dialog, you can tell AutoMagic to use a pop-up window. This will be especially useful for those wanting to turn the info bar off entirely to save screen space. Because the suggestions might not be useful, after several seconds the pop-up window will disappear in order to get out of your way.

The pop-up AutoMagic window is dramatically abbreviated. There are no icons and no hotkeys. It just shows numbers and guesses. It also only shows a maximum of 9 options instead of ten in order to avoid cluttering up the screen too much.

> **Use Numbers:** The View toggles dialog contains a "use numbers" option that determines whether or not the numbers appear to the left of the commands. If you have this option turned on, then the number keys can be used to select options, but the user will have to use the "type in text" function to actually type numbers, just as you would normally do with Hyperkeys for typing things in.

Because AutoMagic can speed up globaling under most circumstances, using the AutoMagic numbers overrides not only the ability to type numbers, but the default Hyperkeys for doing multi-stroke globals (7, 8, 9 and 0.) These commands will be automatically reassigned to Shift+7, 8, 9, 0. You can turn off the "Use numbers" option if you want to re-assign these back to just the digits, but then you'll have to select the AutoMagic options by clicking on them with the mouse, or by assigning keystrokes for the AutoMagic selection menu items under Window/Select. (These selector items default to using Shift+Alt+1, etc.)

> **Hotkeys:** Also, in the View toggles dialog, you can turn off the AutoMagic hotkeys to save even more space on the display. This is good for experienced users who already know all the hotkeys and just like the time-saving suggestions.

It would take many, many pages to describe all of the different types of edits and different commands that AutoMagic will suggest. Instead, a few specific examples will be given to get you oriented to this new and hopefully faster method of doing things.

Sometimes, AutoMagic will suggest a global similar to what you would see in the Global Magic feature. If it does this, and if you are making a longer dictionary entry that does not conflict with anything already in your dictionary, then the AutoMagic options will immediately change to ask you which dictionary the global should go into. If you select none of them, it will stay as a local replace.

You will get different editing suggestions depending on the overall program context and position in the editing document. Here are some example contexts:

- No document on the screen
- In a note file
- In a note file with strokes marked

Introducing Total Eclipse 5

- In a dictionary
- In an empty dictionary
- In a text file
- In a marked block of text
- At the end of a paragraph
- At the beginning of a paragraph
- On a conflict
- On an untranslate
- On a misspelled word
- On a form field
- etc.

There are dozens and dozens of contexts that the system looks at, and it will provide the 10 most likely edits for each context.

Here are some samples showing different types of edits that would be suggested depending on the cursor position. Note that this is just scratching the surface of what it can do. The samples here containing numbers are only a fraction of the AutoMagic edits that take advantage of the number conversion capabilities, for example. In these examples, If the cursor is resting on the text on the left, the text on the right will be suggested

theretical --> theoretical or heretical
is she --> she is
they're --> they are
he is --> he's
forming --> form
are throw SKOPic --> arthroscopic (4)
they they --> they or they -- they
to --> do
three car --> three-car
look over --> look-over
did yes --> did, yes
MR. SMITH (speaker 1) --> MR. JONES (speaker 2)
Q --> Q (By Mr. Smith)
abiding --> a bid in (undoing a Translation Magic error)
correct? --> correct.
winchester --> Winchester
JOHN --> J-O-H-N
varnish --> SRAR TPHEURB
good-night --> goodnight
chair man --> chairman
wherein --> where in
there, is --> there is
yes, he --> yes. He
7 --> seven
twenty three --> \$23
one fifteen --> 1:15

AutoMagic won't always have the correct edit in position 1. That's why there are ten possible choices. Even then, it's impossible to predict every edit that you could have to do, but early statistical results seem to indicate that it can anticipate edits in a job and offer the correct edit among the 10 choices about 4/5ths of the time.

Introducing Total Eclipse 5

For **Voicewriters**, AutoMagic will also offer speech recognition alternate choices directly from the speech engine. This is true for single words or for a marked block. Note that when you select one of the alternates, it is equivalent to a local replace, meaning that if you have "defer corrections" turned on, it will put a correction in the list, and if you have it turned off, it will apply the correction to the speech engine immediately.

Info Bar

The Info bar is a new user interface element. It appears on the left side of the screen and has a number of features which display helpful information. Here are the features of the info bar:

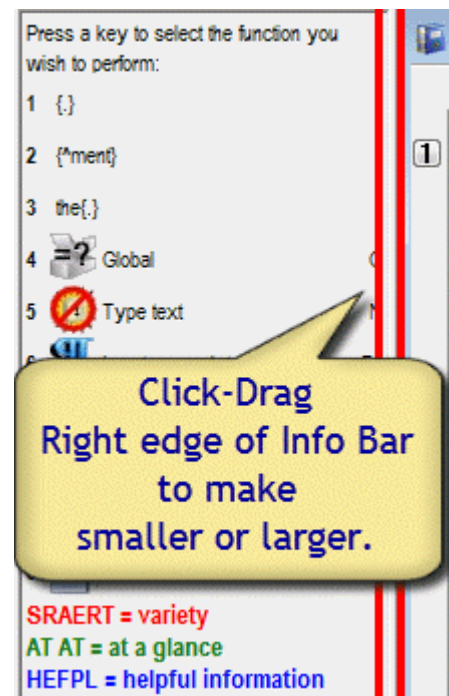
> **Activation:** You can turn the Info bar on or off in the "View toggles" dialog. This dialog appears in two places: Window/View toggles, and User settings/Display/View toggles. Each user settings file will separately remember if the info bar is on or off, so you can have it on for some settings and off for others.



> **Size:** With the info bar on, you will have a bit less room for editing, but that can be addressed in several ways. You can decrease your zoom level in the editing display, and you can also resize the info bar by clicking and dragging the right side of the bar.

Be aware that the text that appears in the info bar changes depending on the context, so you will want to be careful about making it too small. Certain items, such as the auto-briefs, will word-wrap, but others will display text with the ... to indicate that there's more text than there's room for. You can shrink the info bar font if you want to fit more text in less space.

> **Font:** In the view toggles dialog, there's an info bar font setting which will change the font use for all items displayed in the bar except for the auto-briefs, which have their own font setting.



Introducing Total Eclipse 5

> **Translation statistics:** Under window/view toggles, there's an option to have your translation statistics displayed in the info bar.

Note that there's a "setup" button next to the checkbox that allows you to select what items are displayed. This is the same dialog that you get when you right-click on the realtime statistics window.

Putting the statistics in the info bar will make it possible to permanently close the statistics window and get it out of the way.

You can now click and drag the top bar of the waveform display on the info bar to resize it.

Note that the same bar size will be used whether you're displaying the waveform or the level now that it's user-definable.

> **Waveform Display:** When you're recording audio, if you have the translation statistics turned on in the info bar, normally you'll get a level indicator, just as you would on the realtime statistics window.

If you would prefer a waveform display, you can select Window/View toggles, Waveform display, and on the bottom of the info window you'll get a visual display of the actual audio waveform with peaks and valleys.

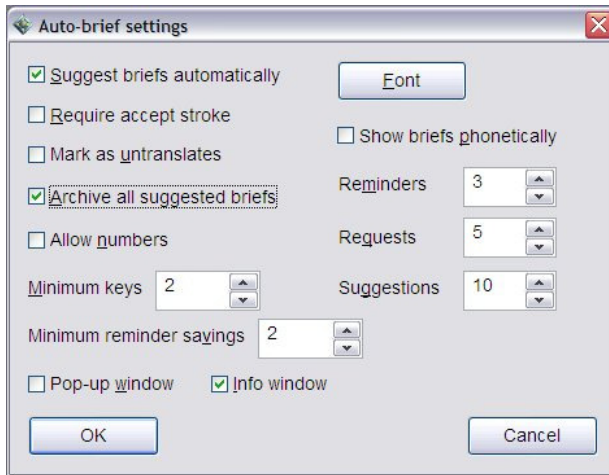
This type of display makes it easier to see if you're getting a faint signal, or if you're getting a very loud signal that's causing distortion.

> **Auto-briefs in Info Bar:** Under window/view toggles, there's an option to display the auto-briefs in the info bar. This same option also appears under the User settings/Realtime/Auto-brief settings, where there is also an option to turn OFF the pop-up auto-brief window that appears over the document.

Since the info bar is so long, if you're using the info bar instead of the pop-up window, you can boost the maximum number of briefs shown quite a bit. Use the settings under User settings/Realtime/Auto-brief settings to do this.

Introducing Total Eclipse 5

Auto-Brief Improvements



> **Auto-brief settings dialog:** Under user settings/Realtime, next to the Auto-brief checkbox, there is now a "Settings" button which brings up the new auto-brief settings.

> **Color-code auto-briefs:** Auto-briefs will now show in three colors which are user-definable under the User settings/Display setup:

Suggestions: Normal briefs suggested by the software automatically

Requests: Briefs that you specifically requested

Reminders: Briefs that you already have in an active dictionary

> **Settings added for separating auto-briefs into three categories**

In the User settings/Realtime/Auto-brief settings, there are now three separate categories: Reminders, Requests and Suggestions. Items in the auto-brief window will now appear in that order.

> **Settings for maximum number for each category:** The items will still be color-coded, but will no longer be mixed together. Also, since each category has its own individual upper limit, a large number of reminders will no longer scroll the suggestions off the screen. Likewise, a large number of suggestions will no longer scroll very important requested briefs off the screen.

> **Font for Auto-brief:** Previously, the auto-brief font would be automatically selected based on the size that the user dragged the window. However, that turned out to be a bit awkward, so the resizing mechanism for the auto-brief window has been removed and replaced with the separate option to select the maximum number of briefs, and this option to select the font and size specifically.

> **Auto-briefs can show as untrans:** If you turn on User settings/Realtime/Auto-brief settings/Mark as untranslates, all auto-briefs will appear in the transcript as untranslates, much like entries used by Translation Magic.

Introducing Total Eclipse 5

> **Auto-brief minimum phonetic elements setting:** The User settings/Realtime/Auto-brief settings/Minimum keys setting allows you to define the minimum number of steno keys in a suggested brief.

The auto-brief function will normally find the smallest possible sequence of steno keys that you are not already using for something else. For example, if you write "teleological" in five strokes, auto-brief is perfectly happy to suggest TE for that if you aren't already using the TE stroke by itself for anything else.

If you don't want it to ever suggest very small briefs, you can specify a minimum number of steno keys that an automatically-generated brief should contain. If you set that value to 4, for example, then it might suggest T*EL instead.

Note that It will try to make a stroke out of the minimum value in terms of phonetic elements AND steno keys.

For example, if you have it set to 4 and you write "goliath" it will NOT simply accept TKPWO as a result, even though it has 5 keys, because it only has two phonetic elements ("go").

Instead, it will start by attempting to phoneticize "goli", which will end up with TKPWHROEU, which will be accepted since it contains at least 4 keys.

Likewise, if you have "allocation" it will start by attempting to phoneticize "allo" but since there are only three unique letters, it will end up with AOL, which is less than four keys, so it will keep extending.

If auto-brief is instructed to (or has to) create a double-stroke brief such as TPHRAT TPHRAT, each individual stroke must meet the minimum phonetic elements guideline indicated by this setting.

> **Auto-brief optional acceptance stroke:** If you want to use the brief-generating features of auto-brief, but you don't want to have the briefs be immediately active, you can turn on the User settings/Realtime/Auto-brief settings/Require accept stroke feature.

If you have that option on, the briefs that appear automatically in the brief window will NOT be immediately available for use, so hitting the strokes indicated will behave normally.

If you then want to be able to accept a suggested brief, you will have to put the following line in your metadictionary:

```
{ABACCEPT}={/?ABA}
```

You can then put an {ABACCEPT} entry in your dictionary to accept the last auto-brief.

Hitting that entry will take the last brief in the list and copy it to your job dictionary, at which point, the brief will be active and will appear in the "Brief requested" color.

If you wish to accept an entry that is NOT the last entry in the brief window, you will have to first remove any suggested entries below it in the window using the {ABREJECT} entry.

Introducing Total Eclipse 5

> **Job-specific auto-brief dictionaries:** Normally, the auto-brief dictionary is cleared at the beginning of each realtime translation. Any briefs that you actually used during the previous session were copied to the job dictionary, but briefs that you didn't use would disappear.

If you wish to be able to go back and review all of the suggested briefs from the previous translation in case any of them might be useful, you can turn on the User settings/Realtime/Auto-brief settings/Archive all suggested briefs function.

With that function on, at the end of each realtime translation, all suggested briefs will be copied to a jobname_AB.dix file for future review. Note that this dictionary will never be automatically included in any translation, nor should it be.

> **Separate auto-brief phonetic option:** The User settings/Realtime/Auto-brief settings/Show briefs phonetically will now be used to determine whether or not auto-brief steno is displayed phonetically.

Previously, this functionality was tied to the User settings/Translate/Phonetic untranslates setting. They are now entirely independent.

> **AB reminders of unused suggestions should go back in suggestions list:** Reminders will now be limited to those briefs and auto-briefs that you have created or used. Any repeated suggestions will simply be placed back in the suggestions list, not in the reminders.

> **Optionally show brief reminders with minimum stroke savings:** Under User settings/Realtime/Auto-brief settings, there is now a setting for the minimum reminder savings. This is the minimum number of strokes that must be saved by a brief in order for the feature to remind you when you've written something the long way.

The default setting is 2. If you want it to remind you if there is any savings at all, set it to 1. You can, of course, set it larger.

> **Optionally only show reminders for commented entries:** If you set the minimum stroke savings to 0 (zero), that doesn't disable the feature, but close to it. What that will do is only remind you of briefs if they are commented in some way.

In other words, if a brief has "AB" in the comment field, or "brief" or anything else at all, it will be eligible to come up as a reminder.

Note that with this setting, any number of stroke savings is valid (it's equivalent to a setting of 1 with an extra criterion requiring the comment.)

> **Requesting a brief should give reminder steno, if applicable:** So if you mark a word or phrase and request a brief, the first thing it will do is see if you need a reminder if that word or phrase already has a brief, and if so, it will remind you of it immediately.

Also, If you specifically request a brief for a three-stroke sequence, even if there's a two-stroke sequence in your dictionary, it will go ahead and suggest a one-stroke brief instead of giving you the reminder which is only marginally better.

Introducing Total Eclipse 5

> Auto-brief is much quicker and better at reminding users of existing briefs

This is actually quite a large improvement. Existing briefs will now show up immediately once you write something the long way, no matter how few words you've written or whether any of those words are on the ineligible words list.

For example, even if you write two separate strokes U = you and -R = are, you will get an immediate "you are = UR" reminder from your dictionary if you have that brief defined (and if you have requested reminders for 1-stroke savings or better.)

The reminder feature will only show one reminder per word or phrase, even if it can find multiple briefs for it in the dictionaries. It will first narrow down the reminder possibilities by stroke and will show the shortest one. If there are multiple entries with the same number of strokes, it will pick the most used one, assuming that that is the correct entry and the rest are probably misstrokes.

> **Auto-brief numbers allowed:** Normally, the auto-brief function will ignore anything containing digits, and will ensure that suggested strokes never contain digits.

If you wish to be able to auto-brief number sequences, turn on User settings/Realtime/Auto-brief settings/Allow numbers.

Note that just turning on this option will not produce ideal results, since writing a long number such as 574,974.23 would result in a suggested steno brief of "57". This could cause big problems when writing a number.

If you intend to use this feature, it is recommended that you supply additional keys to be added to number briefs, as described below:

> **Auto-brief number-specific steno theory:** A line starting with a + sign in the User settings/Programming/Auto-brief steno theory will normally add steno keys to all suggested auto-briefs.

If you enter +EUG, then a suggested brief for "hydrodynamic" would probably be "HEUG".

Because numbers are now permitted, you can optionally add a sequence of steno keys that will ONLY be added to auto-briefs containing numbers. This is indicated by including the number sign # after the + sign.

For example, you might add +#SZ in the auto-brief steno theory. Then, when you write 574,974.23, the auto-brief procedure will be able to suggest 5SZ as a brief for that number.

Note that the number additions and the regular additions are separate, independent settings and you can use only one or only the other or both simultaneously.

> **Auto-brief is able to suggest briefs with numbers such as dates: "May 12, 2007"**

If you have the numbers enabled in the auto-brief function, AND you remove the months from the list of auto-brief ineligible words found under User settings, Programming, then dates in this format will end up with a brief suggested for them.

Introducing Total Eclipse 5

This could either be very handy or very annoying, depending on the circumstances and the types of jobs you do, so experiment and decide what works for you.

> **Auto-brief should not use strokes that are in the slop strokes table**

> **Auto-briefs should detect plurals and suggest singulars:** If you write a long word that is pluralized, such as "hydrodynamics", the auto-brief feature might suggest a brief for it.

If it can detect that it's a pluralized form of a word (in other words, if it sees that "hydrodynamic" is a legitimate word) then it will suggest a brief for the singular version instead of the plural version.

If you have the User settings/Translate/Integral pre-suf feature turned on, then you will be able to write the integral prefix or suffix with the suggested brief. For example, if it suggests H*EU = hydrodynamic, you can write H*EUS to get hydrodynamics.

File Management

> ***New text document file storage system***

Total Eclipse now manages documents entirely in memory, just like dictionaries. The "Open" function loads the whole document into memory, and closing the document saves the whole document back to disk.

The saving/loading process has been changed so dramatically that many functions in the software have been sped up. Block copy and paste a large section of a document and compare how long it takes with old documents vs. new documents. Compare the speed of a Quick Tran (with Translation Magic turned off, since that's the slowest part of the translator!)

Documents are considerably smaller than in previous versions; less than half the size for exactly the same data.

Documents should be much, much more reliable. "Invalid paragraph/optimize document" errors are no longer possible.

> **Compatibility WARNING:** This new document file format is incompatible with previous versions. This new version can load old files, but once they're modified, these files can no longer be opened by any previous versions, much like the transition from 3.X dictionaries to 4.X dictionaries. As with that transition, old-format .ecl files will first be backed up to the username(v3) folder in case of a dire emergency requiring the user to go back to an older version.

Introducing Total Eclipse 5

> 10 Rotating Auto-Backups

The current save/load functions can save or load a 300-page document in under a second, so this new mechanism is fast enough to use a full-document auto-save.

The auto-backup function will no longer make .bak files in the jobs folder. Instead, it will make a series of backup files named

jobname.bk0,
jobname.bk1,
jobname.bk2, etc.,

...up to jobname.bk9, representing the last 10 versions of the file in the backup folder. By default, this is "\EclipseNT backups", but remember that you can change that under User settings/Programming/File locations/BACKUP=etc.) If you have your backup interval set to 30 minutes, for example, it will keep the last five hours worth of backups. This will allow you to go back to an even older version of the file if the most recent backup contains an error that you were trying to recover from (such as a disastrous block-delete or ill-advised text global.)

> **Quick-save every 5 seconds:** Between full saves, every five seconds the software does a quick-save in order to preserve document integrity in the event of a problem.

The upshot of all this is simple: Previous versions saved every paragraph as you edited, so in the even of a crash or a lockup, you would never lose more than the changes in one paragraph. In this version, you cannot lose any more than five seconds of work. Yes, you read that correctly. Five seconds.

> **Revert function should ask which backup file to copy over the .ecl:** When you press the "Revert" button in the file manager it will now display a list of all of the backup files for the current job. The one you select will be copied over the Jobname.ecl in the jobs folder. Note that this does NOT save the original Jobname.ecl file anywhere. You should NOT use "Revert" unless the current version of the file is unsalvageable.

> **Auto-backup the main.dix:** The user's main dictionary will now maintain .db0, .db1, .db2, etc., files in the backups folder just as with jobs. Note that the main dictionary gets saved and backed up each time you exit the program.

> **Auto-backup user.ini files:** The user's current .ini file will get backed up when saving the settings or exiting the program. These backups will go in the backups folder as filename.ub0, .ub1, .ub2, etc.

> **Job info expanded:** The job info in the file manager will now contain additional information.

The information under tools/statistics, for example, is now stored permanently when you run the statistics function and can be seen in the file manager without having to open the job.

The translation statistics are also stored permanently so that they can be reviewed at any time in the future in the statistics window or the info bar, or in the file manager info pane.

In addition to the translation statistics and the calculated statistics from the tools menu,

Introducing Total Eclipse 5

there is other information that is stored, including the name of the last person to edit the file, the name of the person who translated it, and the name of the scopist (the last person to open it on an edit station.)

All of this information is now listed in the file manager in the info pane. In addition, the info pane has been reorganized somewhat to make the information easier to read.

Don't forget that the "search" function in the file manager allows you to search for any job in the current folder that contains the text you're searching for anywhere in the info that appears, so if you do a search for "Frank", for example, that will suffice to find any job that contains "Frank" as a reporter, scopist, speaker or name that appears among the job variables.

Also note that you can include the label in the search, so if you search for "Scoped by: Bob" it will find any job where that specific text appears in the info pane, allowing you to narrow your searches considerably. File information in file manager updated

> **Support the MyDocuments folder in the folder system:** In the User settings/Programming/File locations settings, you can now directly access your documents folder using {DOC}.

For example, if you wanted your Eclipse documents to go in a folder named "Eclipse" as a subfolder of your "My Documents" folder in XP (or your "Documents" folder in Windows 7) you would just use:

```
MAIN={DOC}Eclipse
```

That way, if your jobs folder is

```
JOB={MAIN}Bob
```

then your job files would be in My Documents\Eclipse\Bob

The {MAIN} folder will default to {DOC}Eclipse in this version of Eclipse so that documents will be in your documents folder rather than being a subfolder of the Eclipse program files folder, which Windows Vista and 7 consider a security breach.

Old 4.x documents will remain in the Program files folder if you have a need to use the old version to access old jobs.

> **Folder system should create folders that don't exist:** If you were to edit the file locations and type in a folder that does not exist, such as BLOCK={JOB}court\blocks, the system would automatically create the folder or folders in question as soon as any part of the program needed to access that folder.

> **User migration process added:** Because the new default location for users is under the documents folder, when you run this version of the program you will have no users listed.

However, a new item has been added to the end of the users list to import an old user. When you select this item it will give you a list of all of your user.ini files. Note that you can select multiples by ctrl+clicking, clicking and dragging, or shift-clicking.

Introducing Total Eclipse 5

When you import an old user, it will import all of the settings and will change your MAIN folder to {DOC}Eclipse, storing a new user.ini file in that location and creating a new jobs folder inside it.

It will also copy over your main dictionary, spelling dictionary, phone books, .set settings files, block files folder, and any .ecl files the block files folder.

It will NOT copy over all of your jobs. If you wish to have duplicates of all of your jobs, you are welcome to copy them, but since you can easily run the older version to access the older jobs, this isn't really necessary.

You can easily copy over job files by using the file manager, hitting the "Browse" button, navigating to Program files\Advantage Software\EclipseNT\Users\username, selecting files and selecting "copy to" the (new) Jobs folder.

> **Support use of {PROG} to indicate program files folder:** The file locations currently allows parent paths, such as MAIN=C:\Eclipse and JOB={MAIN}Bob, which would mean that JOB was really C:\Eclipse\Bob.

If you have a need to set up a folder that is defined as a specific subfolder of the location where the program itself is stored, you can use {PROG} even though there's no PROG=etc. line in the list. The Eclipse program folder is determined during the installation routine and is stored in the Eclipse.ini file in the Windows folder.

> **Support path system in the function to execute a script line:** A seldom used function of the script line print command is the execute command, indicated with x | path. This command will open a program or document with the path given. This is the function used by Keith's editing tutorials for running visualizers.

In order to make it easier to deal with installations in non-standard locations, this command will now support typing in {PATH} commands such as :
{PROG}Documentation/Visualizers/name.

Introducing Total Eclipse 5

Editing

> **Spelling errors can now be marked while editing:** Under User settings/Edit/Spell options there's a new checkbox labeled "Display errors while editing". It defaults to being on.

This feature will put a red underscore under all misspelled words.

> **e-book reader proofreading procedures implemented:** This has currently only been tested on a Kindle, but should theoretically work with any e-book reader that uses the MOBI file format.

There is only one setup step for this procedure: Plug in your e-book reader and find the folder where it stores documents using Windows explorer, then set User settings/Programming/File locations EBOOK=etc. to point to that path.

For example, my Kindle is I:\documents, so I have the following:

EBOOK=I:\documents

Then, when you want to proofread documents, use the following procedure:

Tools/Convert/Text files/To/e-book proofreading. Select the files and hit Next. That's all.

Disconnect your e-book reader and start proofreading. On a Kindle, you can move the cursor through the document and mark text and/or start typing on the keyboard to make a note to yourself.

When done proofreading, reconnect the reader and use Tools/Convert/Text files/From/e-book proofreading. Select the .mbp files for the jobs that you've proofread and hit Next.

Having done that, when you re-open the documents in Eclipse you'll find that each mark has turned to "Scopist text" color, which you can scan to using the Scan function. Also, anywhere you wrote a note, you'll have the text color change and the note will appear at the top of the paragraph inside a comment line.

IMPORTANT: Once you have done the file/to/e-book, don't make any changes in the document in Eclipse before you re-import the marks and notes or they might not end up in the right places.

> ***Text globals must now be specifically selected using a checkbox***

This is a significant change, but hopefully a helpful one.

If you mark a block of text before hitting the global command, the system will now assume that you want to perform a normal steno global on the text that you have highlighted.

If you wish to perform a text global, you must check the "text" box in the global window, which always defaults to off.

Introducing Total Eclipse 5

If you are accustomed to doing text globals regularly, this may seem somewhat inconvenient, but there's a trade-off. You can now start a global without first marking the text, using the global dialog to highlight the text you wish to replace, and then check the "text" box in the dialog after making the selection, which could end up being a faster way to perform the global.

> **Form field math formulas to allow for billing pages, etc.:** You can now enter a formula that will determine how a form field is filled in mathematically. The "Add blank" dialog has a whole new box for filling in the formula when creating form pages.

Note that you can also have a variable name for that field so that after the calculation is performed, the result is stored in a variable for later use.

The mathematical formula parser understands the following operators:

- + addition
- subtraction
- * multiplication
- / division
- () precedence. Example: $(7 - 2) * 3$ produces 15, but $7 - (2 * 3)$ produces 1
- % modulus (remainder). NOT percentage!

It also understands the following functions, though you probably won't need them:

Abs - absolute value	Sqrt - square root
Log - natural logarithm	Log10 - base-ten logarithm
Sin - sine	Cos - cosine
Tan - tangent	ASin - arcsine
ACos - arccosine	ATan - arctangent

For example, `Abs(4 - 10)` produces 6

I realize that for the most part, you're going to be using this on billing pages and for scopists invoices, etc., and you won't need the advanced math, but it was already in the parser, so I left it in.

Who knows, someone might find a use for some of these functions. For example, you can do calculations for compound interest without an iterative function using logarithms. Remember your calculus? (Can you charge an attorney interest for paying transcript fees late?)

Finally, the formula parser can use variable names in place of numbers, which includes any variables you have created and filled in in the job. For example, if you have created a form field containing a variable `PAGERATE` and filled 2.75 into that field, you can use `PAGERATE` in a formula calculation.

The following variable names are built-in, and will be filled in automatically with the appropriate value:

- PAGES - the total number of pages in the document
- PAGE - the current page number at the cursor position
- YEAR, MONTH, DAY, HOUR, MINUTE, SECOND -- elements of the current date and time

Introducing Total Eclipse 5

An example of use:

Let's say you want to calculate a bill for the total number of pages in a transcript minus the form pages.

On the last page of your appearances, you put a comment line containing a formfield with a variable FORMPAGES, and a formula containing only PAGE. This will calculate and store the page number of the last form page before the content starts.

At the end of the file, you have a billing page with form field containing a variable PAGERAGE which you fill in manually.

In the invoice, you have a total bill form field containing the formula:

$(\text{PAGES} - \text{FORMPAGES}) * \text{PAGERATE}$

> *name_@email.com now working after a + sign*

You can now put name_@email.com anywhere in a list item, even if it's not the first field contents.

> *Find steno in a job now finds macro strokes:* Macro strokes are "invisible" in the document, though you can see them in the steno window. When doing a steno search, the system will now attempt to find the strokes in the visible portion, but if that fails, it will look for invisible strokes.

Note that this will find deleted strokes and macros strokes, both of which are invisible. Also note that the cursor cannot land on these strokes, so the cursor will simply get as close as it can if it finds a match.

> *Find function should support hidden characters:* The hidden characters are rather cryptic, and most times you don't need to look at them. However, there are times when it could be useful to be able to find a particular sequence of hidden characters.

You can now find hidden characters by using the same syntax that is used in the hidden text dialog, such as <2>.

Here's a basic rundown of the hidden character codes:

<1> indicates a steno stroke for the purpose of steno tracking

<2> indicates a font change

<3> indicates a change of text type

<16> lockspace

<17> tab

<21> literal-case on

<22> literal-case off

The values 9-15 are reserved for storing values that follow the first three commands (steno/font/type). For steno, it indicates the number of strokes. For the font, it indicates the

Introducing Total Eclipse 5

font number. For the text type, each number represents a different type of text.

Here are what the values mean, with the text type given in parentheses. Note that you use these in conjunction with the other hidden characters. For example, searching for <3><11> would search for a user-selected conflict, and <2><14> would search for a font change to font number 6 (found in the User settings/Document/Advanced/Master font table.)

<9> - 1 (untran)
<10> - 2 (unselected conflict)
<11> - 3 (user-selected conflict)
<12> - 4 (computer-selected conflict)
<13> - 5 (typed-in text)
<14> - 6 (scopist text)
<15> - 7 (automatic punctuation text)
<8><9> - 8 (form field)
<9><9> - 9 (redacted)
<10><9> - 10
<11><9> - 11
<12><9> - 12
<13><9> - 13
<14><9> - 14
<15><9> - 15
<8><10> - 16
you can calculate larger values thusly:
<8-15><10> - 16-23
<8-15><11> - 24-31
<8-15><12> - 32-39
etc.

For example, if you had a long font list, and you wanted to search for a font change where the text was being changed to font number 29, that would be <2><13><11>

> Document utility functions split into separate commands: The "Document utility" function has been removed from the menus and replaced with a series of commands instead. These commands have also been added to the list of commands available for assignment in the edit keyboard function.

Format / Text type -- Ctrl+Shift+R
Format / Quote -- Alt+Shift+Q
Format / Unquote -- Alt+Shift+N
Tools / Realtime / Format script -- Alt+F12

The old keystroke for the document utility is still in place so that older macros will still work, but users are encouraged to use the new commands.

> Steno window shouldn't center active stroke when in realtime leaving lots of extra space below

When in realtime, the highlighted steno stroke will now be closer to the bottom of the screen to give you more steno context above the highlight.

Introducing Total Eclipse 5

Translation and Realtime Transcription

> Integral prefixes and suffixes apply during translation

Under User settings/Translate there is now an option called "Integral pre/suf".

This feature will cause the translator to attempt to apply the integral prefixes and suffixes during translation rather than only using them during globaling or adding a dictionary entry.

For example, if you have "EBGS PERT = expert" in your main dictionary, you can write EBGS PERTS and get "experts" even if that definition is not in your dictionary.

Note that it does require that the resulting word be a correctly spelled one or it won't apply the integral prefix or suffix.

To review, here are the rules for the integral prefix/suffix definitions in the User settings/Programming tab:

wordpart=NewStenoKeys,ExistingStenoCheck

wordpart can be either a prefix or suffix: ^ed or ^ing or dis^ etc.

NewStenoKeys are the keys that get added to the stroke to make an integral prefix or suffix, such as -D for ^ed.

ExistingStenoCheck has two possible syntaxes: ExclusiveSteno or +InclusiveSteno.

Some elements should NOT be applied if certain letters are already in the stroke.

For example, you should NOT apply a -G = ^ing if there is already an -S in the stroke. That's why the ^ing entry looks like this: ^ing=-G,-GTSDZ

That means as long as NONE of the letters -GTSDZ are in the stroke, it's OK to add the -G and ^ing to the end. That's the EXCLUSIVE steno.

Other suffixes or prefixes should be applied ONLY if certain letters are already in the stroke. For example, in the tucked-in {^er}, users want to be able to specify that they ONLY tuck in an -R before the letters G, T, S or D. That's why the ^er entry looks like this: ^er=-R,+-GTSD

The + sign indicates INCLUSIVE steno that must be present for that suffix to be applied

> Text replacement dictionary entries will now apply to multiple entries

This is a subtle but big and important change.

Previously, though it was possible to make a text global and dictionary entry such as ["discover corporation"]=Discover Corporation], it was limited.

Such globals would only work during the current editing session and would not apply to future translations or to a continuing realtime translation. That's because the translator would only

Introducing Total Eclipse 5

apply text globals to individual dictionary entries, not to sequences of text made up of multiple dictionary entries.

This version is capable of going back and text replacing multiple entries as long as the text pattern matches, so entries such as the example above are now considerably more useful.

The one limitation is that it replaces these retroactively during the translation process in order to take advantage of the complete translation parser to produce the most accurate text possible. Because of this, the realtime output has one minor limitation.

If you're outputting to a litigation support system, there's no problem. It will simply backspace over the incorrect text and output the correct text. If you're doing closed captioning or CART (using the CART window) and if you have the "allow backspaces" option turned off, then you are prohibiting the system from sending backspace characters.

In this case, you can still have the "flush word delay" set to a non-zero value and it will at least be able to text-replace the last word, even if it's made up of multiple dictionary entries. I bring this up because some captioners have said that they had, in older software, been able to make spur-of-the-moment text entries such as "DDD=Discover Corporation" and simply written their glue alphabet letter for D three times to get the replacement. That technique will now work in Eclipse.

> **Automatic [Month(ordinal), year] format for number conversion:** There is a new built-in date format for the number conversion that can be used by the translator which handles a sequence of undifferentiated digits following a month name.

Note that in order to use this, you will have to change your metadictionary entries for your month names to use /"DY or /"DO instead of /"N following the month name.

If you use DY, it will use "day, year" format, such as "May 3, 2010." If you use DO, it will use "ordinal day, year" format, such as "May 3rd, 2010."

Using this format, you can write a sequence of anywhere from 1 to 6 digits following a month name, and the system should determine what format you want based on the number of digits and the contents:

X --> May X

XX --> May XX

XXX --> May X, 'XX

XXXX --> if XXXX starts with 19 or 20, then May AABB, otherwise May AA, 'BB

XXXXX --> May X, XXXX

XXXXXX --> May XX, XXXX

> **Continuation paragraph automation for doubles:** There's a new option under User settings/Translate to "Suppress double paragraphs."

This will convert any double paragraph label into a simple continuation paragraph.

In other words, if you are already in a Question or question continuation paragraph, then hitting a {Q} will change it into {N} automatically.

Introducing Total Eclipse 5

Likewise, hitting {A} will be changed to {N} if you're already within an answer, and hitting {S:Name} will change to {N} if you write a {S:Name} entry with exactly the same name in it.

> **Serial input passthrough feature created:** This was created for captioning companies that need to use a mix of realtime and scripted captions, though it would theoretically work for any device or realtime text delivery system that used serial ports. It allows a coordinator with an Eclipse edit station to send scripted captions while also receiving a feed from a captioner through a serial port and switching back and forth between the script and the feed.

Here's how the new feature works. You set the User settings/Input/Writer type to "none." Select COM port and hit the Setup button to configure the input port for the non-writer. This is the port that you will be able to receive data on and pass it through to the outputs.

In order to toggle on the port pass-through, go to Tools/Edit toggles and turn on the new checkbox that reads "External COM port input." Then hit OK to close out of the dialog. Note that you can easily write a macro to accomplish that toggling function in a single step.

If you have the closed captioning preview window open, you will see a (External input enabled) message that will appear whenever that feature is turned on.

When that feature is on, any data coming in through the input port will be sent to all of the outputs directly. Note that means that if you are hooked to two or more encoders, they will all get the same data sent to them. When this feature is off, all data coming in through that port will be ignored.

When transitioning between pre-captioned material and realtime, the system will always attempt to break at a linefeed, or to insert a linefeed so that there will always be a linefeed between them.

> **Bridge should support text globals:** Performing a text global in Eclipse will now send that global to the Bridge stations if you have apply edits checked.

> **CART window spacing option:** Many users have asked for an option in the CART window to double-space. Let's do better than that. When you right-click on the CART window, this version has a spacing percentage. 100% is single-spaced, but you can put in any number from 50% to 300%.

> **Set CART window to be "always on top":** The CART window should always be the topmost window.

> **Division interval changed to use exact times instead of even fractions of an hour**

Some clients who use the division interval were more interested in making sure that the intervals were of a uniform size rather than having them appear as nice, round numbers. Also, this change allows larger intervals such as 40 minutes, even though it's not an even fraction of an hour.

If your organization uses division intervals and would prefer the previous implementation that forced the intervals to stop at even fractions of an hour such as 9:15, 9:30, 9:45 rather than 9:17, 9:32, 9:47 (if the job was started at 9:17) then set the division interval to a negative number, such as -15. It will use a division interval of 15 minutes, but will break the divisions

Introducing Total Eclipse 5

on even fractions of an hour, even if you started the job at 9:08. If you want divisions at 9:08, 9:23, 9:38, etc., use 15 instead of -15.

> Edit word buffer should refer to number of words before cursor if wait for cursor is specified: Previously, the edit word buffer and the "wait for cursor" options would wait for the number of words specified OR two words behind the cursor, whichever was longer.

In practice, when using the "wait for cursor" option, the cursor would almost never advance far enough to make the edit word buffer value relevant.

In this version, if you have "wait for cursor" on, the edit word buffer value now refers to the number of words behind the cursor that the software will wait before sending text to the output system.

> Slop strokes table can now work with partial matches

If you put in a slop stroke of STKPWR=STKPWHR, it will only replace that exact stroke, though it will work in dictionary entries containing multiple strokes.

If you put in an entry of &STKPWR=STKPWHR, it will replace that key sequence anywhere it appears in any stroke, so even a single stroke STKPWROBG would be replaced with STKPWHROBG.

Use this enhancement carefully. If you were to put in &RBS=RBGS to fix misstroked commas, you would never again be able to write word like "curbs" or "suburbs" because ALL strokes containing RBS would be changed to RBGS.

Audio

> Audio recording sent through StenoLink connection

WARNING: If you are going to transmit audio through StenoLink, it is highly recommended that you use a heavily compressed audio format such as GSM or Speex. PCM files are likely to be much too large to stream effectively.

Here's how this works. Instead of having the audio be on-demand through the IP connection, the reporter's computer sends ALL of the audio as it's recorded. The Scopist's computer then ends up with a complete WAV file being updated every few seconds as the reporter's computer records it. Since it's a local copy of the file, the scopist's WAV file is even available after the scopist disconnects.

Currently, there's no handshaking or re-sending of lost data, so if the scopist computer misses something, it's missed. You can always get a complete copy of the WAV file from the reporter later. If the reporter starts recording for two minutes and then the scopist hooks up, they will have missed two minutes of audio (and steno strokes.)

Ultimately, this functionality will be much further refined using a server on the Internet to manage the connections and allow for more complete updating. This feature is intended as a proof-of-concept to test the transmission of WAV file data live over the Internet between the reporter and scopist.

Introducing Total Eclipse 5

> **Back up whole WAV file incrementally to division folder when recording:** This could be an enormous boon for reporter/scopist combinations.

Internet-based automated file repository systems such as DropBox and SugarSync will automatically synchronize files from a selected folder to the Internet, where other people can retrieve the files.

Reporters have been using this with the division intervals feature to give segments of files to scopists.

However, using this system has historically not worked with WAV files since they're being recorded, and these programs won't touch a file that's open.

This version will make a copy of the WAV file to the DIVISION folder after the division interval expires. Subsequently, at each division, Eclipse will update that WAV file by appending the new data to the end of it.

The repository systems already use a system whereby they only transmit the changes to the file, so this will allow the WAV file to be incrementally updated rather than having to send the entire file over and over again, and without having to break the WAV file into many separate parts.

Rather than create a separate option for this feature, what the translator will do is copy the WAV file to the DIVISION folder if you do NOT have a WAVREC line in your file locations. In other words, if you have already demonstrated that you have a system for storing the WAV file in a separate location, it will leave it alone. If not, it will perform the copy every time the division interval expires.

A reminder: If you're going to use this, make sure you include the underscore in your filename so that the divisions will synchronize to the single WAV file.

> **"Sync audio here" button:** It seems so simple. I really should have thought of this before. When using a WAV file recorded on an external source, the timecodes are usually self-consistent, but not aligned with the document's timecodes.

While it's possible to get them aligned, it's a bit of a chore to figure out the correct timecode offset. Now, if you can hit "play" and hear the audio and you can identify where the audio is playing, you can synchronize the file with one button.

Go to a point in the document that plays audio and hit the play button. Move the cursor forward in the document past where it's playing and put it on a spot you'll be able to recognize. Go to User settings/Document/Timecodes.

As soon as you hear the word the cursor is on, hit the "Sync to cursor" button. That calculates a new offset for you automatically and the audio will now be synchronized at the cursor position.

Introducing Total Eclipse 5

Dictionary management

> **Dictionary builder should have steno stroke for back button:** For now, the dictionary builder relies on several pre-defined strokes in the user settings for its operation since it doesn't have its own settings dialog.

If you set up a "stitch mode" stroke, you can use that stroke during dictionary building instead of hitting the "back" button to return to the previous entry.

The advantage of being able to return to the previous entry is that you can enter additional steno for the same word if you wish to have several ways of writing it, which is especially helpful when using it to build job dictionaries.

> **Advanced button in dictionary entry editor should go to master speaker table if user is on a speaker entry:** When working on speakers in a dictionary, it's helpful to be able to go right into the main dictionary and add the speaker to the master speaker table, as well.

> **"Copy" button for steno emulator:** Many dictionary entries are created by using double strokes. When creating or searching for an entry like this, you can now hit just one stroke in the steno emulator and use Alt+C or hit the Copy button to duplicate the first stroke.

> **When editing a macro, search for existing dictionary entries:** Under User settings/Edit/Macros, when you edit a macro you have the option of setting up keystrokes or dictionary entries for it.

If you hit the "Dict Entry" button, the steno emulator will now appear pre-loaded with the steno used for the existing dictionary entry in your main dictionary for that macro as a reminder that you already have an entry for that item.

Note that changing this steno does NOT remove the existing entry. It allows you to create an additional alternative way to execute the macro, just as adding a new keystroke doesn't remove the existing keystroke.

> **Dictionary regex filter should remove hyphen from steno after getting it from steno dialog:** If you use the steno keyboard to search for entries ending with DZ, for example, it should now find ALL entries ending with DZ, such as WEDZ, not just ones ending with -DZ such as HR-DZ.

> **Dictionary should print just the marked block, if applicable:** If you mark entries before you select the print function, it will print only the marked entries.

> **Spell-check one word function should work in dictionary:** You can now use Alt+S to spell check a single dictionary entry. Note that this only works for entries that consist of only one word. Anything else will be met with a "that is not a word" message in the status bar.

Correctly spelled words show "word is correctly spelled" in the status bar, and incorrectly spelled words will bring up the spell check suggestions dialog.

Introducing Total Eclipse 5

> **Dictionary searches should not be case sensitive:** If a dictionary search contains no capital letters, the search will now be case insensitive. Any capital letters will cause it to do a case-sensitive search.

For example, if you do a dictionary search for "bank" it will find both "banker" and "Mr. Banks" but if you search for "Bank" it will only find "Mr. Banks"

> **Add support for the new palantype writer with the * key:** This key can now be used just as the * key on a steno machine, though to eliminate conflicts with older dictionaries, the * key will appear at the END of the stroke on the display, rather than in the middle.

User interface

> **Allow multiple copies of Eclipse at the same time:** It was previously disallowed to run multiple copies of Eclipse. However, the newest version doesn't leave any shared resources open during execution, so it's theoretically possible to run multiple copies.

This does NOT mean that you can have the same document or dictionary open in multiple copies. Previously, that would crash. Now, it will allow a document to be opened in multiple copies, but the LAST one closed will have its changes saved over the other one's version.

Because of this potential, the program will actually ask you if you want to have multiple copies opened. If you answer no, it will do as it did before and bring the active version to the front.

If you answer yes, you will be permitted to run another copy of Eclipse on the same computer. If you're careful, such as if you have multiple users, you could run multiple separate translations, or even multiple separate realtime sessions, depending on whether you had sufficient hardware resources. Any task which Eclipse could normally do only one of at a time can now be done simultaneously by multiple copies of Eclipse running at the same time.

Use judiciously.

> **Show current color selections in Display tab:** Under User settings/Display/Color selections, the actual list box items will appear in the same colors as the selections themselves, making it easier to review color selections and decide what needs to be changed.

> **Put note file name below file list on read notes dialog:** The workflow on the read notes dialog is normally to select the file you want to read, preview it, then fill in a name for the job and hit the Read button.

The controls on the read notes dialog have been rearranges slightly to reflect this workflow so that you work naturally from top to bottom.

> **Remove "Fix" button and V8 check function from globaling dialog:** There just aren't that many V8 users converting over to the new version anymore, so that function has been disabled.

Introducing Total Eclipse 5

- > **Close file manager when using "open" button:** When you open a file using the file manager, the file manager will now close.
- > **Allow user to click on Global Magic results:** Some people just like mousing.
- > **Select/de-select all option for conversion wizard:** All wizard dialogs that show a list that can accept multiple selections will now show a select/de-select all button.
- > **Show an indicator at the top of the paragraph bar if the paragraph starts above the screen:** If the top of the screen is the middle of a paragraph, the paragraph button in the margin will still appear to let you know what sort of paragraph it is (and to allow you to change it, if desired.)
- > **Put user name on lesson player results printout:** This will make it easier for schools to keep track of which lesson results belong to which students when several of them spool off the same printer.

Writer interfacing

- > **Put in "start from which stroke" message for Passport:** You will now get the option of hitting first/last/other when starting realtime on a Passport (unless you use the instant realtime button.)
- > **Lightspeed and Gemini Revolution added to writer list:** This doesn't currently do anything different, but it does allow for future expansion when special features of these writers may be implemented, requiring a specific selection.
- > **Diamante added to list of writers:** You can now select the Diamante from the list of writers on the input tab.
- > **Diamante USB realtime interface added:** The Mira-style USB interface to the Diamante is now connected. Unfortunately, this only works in realtime. To read notes, it is still necessary to plug the SD card into the computer or into an SD card reader. The new note file format is sufficiently different that the USB interface will require some as-yet-undetermined modifications in order to read the notes over the cable.
- > **Diamante/Mira/Fusion native note file format interfaced:** It is no longer necessary to use the compatibility mode on the Mira or Fusion, nor is it necessary to use the conversion utility for the Diamante notes.

Eclipse can now directly read the new date/time style filenames from the Diamante, Mira and Fusion. This includes the time of day and relative timecodes.

- > **Diamante/Mira/Fusion audio sync adjusted to account for auto-pause:** The timecodes read from the new note files and the jobname.apr files allows Eclipse to synchronize with the WAV files recorded on the Diamante, Mira and Fusion, even if you have the auto-pause feature turned on.

Introducing Total Eclipse 5

Indexing

> **Regular expressions added to automatic indexing:** It is now possible to generate an automatic index with absolutely no preparation prior to the translation and editing of the transcript.

Normally, an index line in the transcript takes the following form:

INDEX | Item | Format | Body | Location

Copies of index lines in this format are placed immediately above the paragraphs where the indexable content appears, so normally they will be placed in block files that are used for exhibit parentheticals, examination markers, etc.

There is now a sixth and final item that can optionally be added to an index line, and instead of placing many copies of this index line in the transcript, you only need one:

INDEX | Item | Format | Body | Location | RegularExpression

Index lines in this format should be placed near the beginning of the transcript, after where you place the sorting lines, such as EXAM | ALPHASORT

A regular expression index line is identical in every way to a normal index line. It follows this simple rule:

Any paragraph in the document that contains text that matches the regular expression search will behave AS IF it was preceded by the index line associated with the regular expression.

> Example regular expression index line:

EXBT | \1 | Index1 | | {TR:1}{P} | Exhibit (\d+)

The regular expression is "Exhibit (\d+)", which means to match any pattern consisting of the word Exhibit, followed by a space, followed by a series of at least one digits (at least one.) This would match a paragraph containing "Exhibit 3" or "Exhibit 27", for example, but not "exhibit 3" or "Exhibit doesn't", etc.

For EVERY paragraph in the document that contains that pattern, it would behave as though that paragraph was preceded by an index line containing EXBT | \1 | Index1 | | {TR:1}{P}, which means that it would add an Index1 paragraph to the EXBT index, with the page number on tab stop 1. Note that the way this expression is written, it would work even in the body of a question or answer, for example. If you ONLY wanted it to index the exhibit where it appeared inside a parenthetical, you could use something like \(. *Exhibit (\d+)

Special note: The index item in this example is \1, which is a backreference. Both the index item and the body can contain backreferences which will fill in the appropriate part of the matched pattern.

Introducing Total Eclipse 5

Special syntax: There are some rare occasions where you need to search for a paragraph break as part of a pattern. If you must do this, use {N} to indicate ANY paragraph break. Also, do NOT include any spaces before or after it, because paragraph indentations do not count as spaces. Paragraph labels DO count in a regular expression search, so you can use them to search for and extract speaker names, etc.

Here are a few more examples:

EXBT | No. \1 | Index1 | -- \2 | {TR:1}{P} | \ (Thereupon, Exhibit (\d+), (.*) , was .* \)

This sample would take any parenthetical in the following format:

(Thereupon, Exhibit 2, A contract between Mr. Smith and Mr. Jones, was marked.)

and convert it into an index paragraph like so:

No. 2 -- A contract between Mr. Smith and Mr. Jones 63

**EXAM | \2 -- \1 | Index1 | | {TR:1}{P} | (DIRECT | CROSS | RE-DIRECT | RE-CROSS)
EXAMINATION{N}BY (.*) :**

This sample would take the following sequence:

DIRECT EXAMINATION
BY MR. SMITH:

and turn it into an index paragraph like so:

MR. SMITH -- DIRECT 12

and would ALSO take the following sequence:

CROSS EXAMINATION
BY MR. JONES

and turn it into an index paragraph like so:

MR. JONES -- CROSS 53

Keep in mind that you can use as many different regular expression lines as you wish, because each one will be applied to ALL of the paragraphs in a document.

That means you can use them to generate multiple indexes easily (such as one chronological and one alphabetical) or to easily subdivide indexes according to context. For example:

**PLEXBT | \1 | Index1 | | {TR:1}{P} | Plaintiff's Exhibit (\d+)
DFEXBT | \1 | Index1 | | {TR:1}{P} | Defendant's Exhibit (\d+)**

Introducing Total Eclipse 5

These two lines would ensure that plaintiff's exhibits and defendant's exhibits were placed in two separate indexes, assuming that the parentheticals contained those strings, such as (Thereupon, Plaintiff's Exhibit 5 was marked.)

Keep in mind that with regular expressions you can make the patterns as flexible as necessary. For example, the expression Plaintiff's? Exhibits? ([A-Z0-9\-\-]+) would find "Plaintiff's Exhibit 23" or "Plaintiffs' Exhibit A" or "Plaintiff's Exhibits 24-27"

Here is some further reference material for regular expressions:

Single character commands

- . (period) Represents any single character. "d.g" returns both "dig" and "dog". If you need to search for an actual period, use \.
- \w Represents any word character
- \W Represents any non-word character
- \d Represents any digit
- \D Represents any non-digit
- \b Represents a word boundary. "for\b" finds "for"/ "for this"/ "for?" but not "forget".
- [abc123] Represents one of the characters that are between the brackets.
- [a-e] Represents any of the characters that are between a and e.
- [a-eh-x] Represents any of the characters that are between a-e and h-x.
- [^a-s] Represents any character that is not between a and s.

Number commands

- * Finds zero or more of the item preceding the "*". "Ab*c" finds "Abc", "Abbc", "Abbbc", etc., but since the number can be zero, it will also find "Ac".
- + Finds one or more of the item preceding "+". "AX.+4" finds "AXb4", not "AX4".
- ? Finds zero or one of the item preceding the "?". "Texts?" finds "Text" and "Texts" and "x(ab|c)?y" finds "xy", "xaby", or "xcy".
- {2} Defines the exact number of times that the item preceding the opening bracket occurs. "tre{2}" finds "tree".
- {1,2} Defines a range of times that the item preceding the opening bracket can occur. "tre{1,2}s" finds parts of both "trees" and "trespass".
- {1,} Defines the minimum number of times that a character can occur. "tre{2,}" finds "tree", "treee", and "treeeee" but not "treated".

Every regular expression is a sequence of character commands optionally followed by number commands. If a character command appears without a number command following it, the assumption is that it should appear exactly once. So "Exhibit [A-Z]," will find Exhibit A, but not Exhibit AA. Since the + means "one or more" the expression "Exhibit [A-Z]+," will find Exhibit AA, Exhibit BBB, Exhibit WXYZ, etc.

Introducing Total Eclipse 5

Special commands

- () Grouping. This defines the characters inside the parentheses as a reference or a group. For example, "a(bc)?d" finds "ad" or "abcd"
- | Either/or. Finds the terms that occur before or after the "|". For example, "this|that" finds either "this" or "that". Note that you can have a pattern that contains a partial either/or option by using the parentheses. For example, "Mr.|Mrs. Smith" would *not* search for "Mr. Smith" or "Mrs. Smith". Instead, it would search for "Mr." or "Mrs. Smith". The correct pattern would be "(Mr.|Mrs.) Smith"
- ^ Only finds the search pattern if the pattern is at the beginning of a paragraph. For example, ^Okay would find the word "Okay" only if it appeared at the beginning of a paragraph.
- \$ Only finds the search pattern if the pattern appears at the end of a paragraph. "Objection\$" would find the word "Objection" only if it appeared at the end of a paragraph. The begin/end commands can work together. "^Okay\$" would find any paragraph that contained nothing but the text "Okay"
- \ Search interprets the special character that follows the "\" as a normal character and not as a regular expression. "yes\\" specifically finds "yes?", not "yes" or "ye" (which would happen if you searched for "yes?" because the ? means "zero or one.")
- \1, \2, \3 Represents a backreference to previously matched sequences in parentheses. The expression "b(o|e)\1t" finds "boot" or "beet"

Backreferences in regular expressions

Backreferences allow you to take matched sequences of text and re-use them either in the search pattern or in the replacement. The number of the backreference indicates which sequence in parentheses you want to re-use.

For example, if you search for (Plaintiff's|Defendant's) Exhibit ([A-Z])

It will find either Plaintiff's or Defendant's, then Exhibit, then any single letter from A - Z. Once it finds that string, the backreference \1 will either contain the Plaintiff's or Defendant's, and the backreference \2 will contain whatever Exhibit letter was found.