

E-Prime REFERENCE GUIDE

Walter Schneider*, Amy Eschman and Anthony Zuccolotto

Psychology Software Tools, Inc.

*Learning Research and Development Center, University of Pittsburgh

With assistance from:

Sara Burgess, Gary Caldwell, Brandon Cernicky, Debbie Gilkey, Jennifer Gliptis,
Valerie Maciejczyk, Kimberly Rodgers, and Jeff Szuhay

Copyright 2002 Psychology Software Tools, Inc.
All rights reserved.

Preface

The goal of developing the E-Prime suite of applications is to provide a common, standardized, precise, computer research language for psychology that can be used on today's technologically advanced computers. The E-Prime suite is designed to allow rapid development of experiments that can be run with precision on computers around the world. A common computer language enables researchers in different universities to communicate and share experimental procedures and data. The system must be flexible enough to allow most psychological research that can be run on computers to be implemented. It must also provide precision for accurate data analysis, and even more important, internal auditing to enable the researcher to report the true precision of the experiment. It is beneficial for the research community to have a dedicated staff of experts interpreting and harnessing the rapidly changing computer environments to allow precision experimentation on standard, commercial machines. The E-Prime suite was designed to be able to be learned rapidly given the constraints of precision and flexibility of experimental procedures. E-Prime is designed to match the way an experienced investigator structures and organizes an experimental paradigm. There are now thousands of investigators that use E-Prime for research on a daily basis to more effectively do high quality computer based experimental research.

Walter Schneider
Pittsburgh, PA, USA

Acknowledgements

The development of E-Prime has involved over twenty person years of effort. Many people have contributed to the effort that has involved designing, running and testing nearly a million lines of code. The **initial design team** included Walter Schneider, Anthony Zuccolotto, and Brandon Cernicky of Psychology Software Tools, Inc. and Jonathan Cohen, Brian MacWhinney and Jefferson Provost, creators of PsyScope. The **lead project manager** was Anthony Zuccolotto until the last year, when Brandon Cernicky assumed that role. The **lead programmer** and manager of the project was Brandon Cernicky, who was in charge of all aspects of the Version 1.0 software development effort. Specific **project programmers** include Anthony Zuccolotto (E-Run); Caroline Pierce (E-Merge, E-DataAid, E-Recovery); Jefferson Provost (E-Run and internal factor architecture). The **documentation management** effort was led by Amy Eschman and Valerie Maciejczyk. James St. James, Brian MacWhinney, Anthony Zuccolotto, and Walter Schneider provided editorial assistance and drafted sections of the manual. **Copy editing** was headed by Amy Eschman and Valerie Maciejczyk, assisted by Debbie Gilkey, Sara Burgess, Kimberly Rodgers, Jennifer Gliptis, and Gary Caldwell. Version 1.0 testing involved Brandon Cernicky, Anthony Zuccolotto, Amy Eschman, Debbie Gilkey, Sara Burgess, Jennifer Gliptis, Gary Caldwell. **Sample experiments and Help files** were created by Debbie Gilkey, Kimberly Rodgers, Amy Eschman, Sara Burgess, Brandon Cernicky. During the lengthy Beta Program, Amy Eschman, Debbie Gilkey, Sara Burgess, Brandon Cernicky, Anthony Zuccolotto provided **technical consulting** and dealt with thousands of beta reports. Government grant support in the form of SBIR grants from the National Science Foundation (Grant #III-9261416 and DMI-9405202) and the National Institutes of Health (Grant #1R43 MH5819-01A1 and 2R44mh56819-02) covered significant research on code infrastructure, timing, and precision testing. Partial support came from grants provided by the Office of Naval Research (Grant #N0014-96C-0110) and the National Institutes of Health- NIMH (Grant #1R43 MH58504-01 and 2R44 MH58504-02) for research on biological extensions to the system.

Reference to E-Prime in Scientific Publications

It is important to describe the tools used to collect data in scientific reports. We request you cite this book in your methods section to inform investigators regarding the technical specifications of E-Prime when used in your research.

Schneider, W., Eschman, A., & Zuccolotto, A. (2002) *E-Prime User's Guide*. Pittsburgh: Psychology Software Tools Inc.

Schneider, W., Eschman, A., & Zuccolotto, A. (2002) *E-Prime Reference Guide*. Pittsburgh: Psychology Software Tools Inc.



Table of Contents

Chapter 1 : E-Studio	1
1.1 About the E-Studio Application	1
1.1.1 The Menus	1
1.1.1.1 E-Studio Menus	1
1.2 About the Interface	8
1.2.1 Toolbox	8
1.2.2 Structure View	10
1.2.2.1 Collapsible Tree View	11
1.2.2.2 Renaming Objects	11
1.2.2.3 Deleting Objects	11
1.2.2.4 Moving Objects	12
1.2.2.5 Copying Objects	12
1.2.2.6 Opening Objects	12
1.2.3 Properties Window	12
1.2.4 Attributes	13
1.2.5 Browser	14
1.2.5.1 Sorting Objects	15
1.2.5.2 Deleting Objects	15
1.2.5.3 Cut, Copy, and Paste	15
1.2.6 Script Window	15
1.2.7 Output Window	17
1.3 E-Objects	18
1.3.1 Standard Property pages	19
1.3.1.1 Frame Tab	19
1.3.1.2 Duration/Input Tab	21
1.3.1.3 Sync Tab	29
1.3.1.4 Logging Tab	29
1.3.1.5 Notes Tab	31
1.3.2 Experiment Object	31
1.3.2.1 Overview	31
1.3.2.2 Interface	31
1.3.2.3 Property Pages	32
1.3.3 Procedure Object	43
1.3.3.1 Overview	43
1.3.3.2 Interface	43
1.3.3.3 Property Pages	44
1.3.3.4 Properties Window	45
1.3.4 List Object	45
1.3.4.1 Overview	45
1.3.4.2 Interface	46
1.3.4.3 Property Pages	48
1.3.4.4 Properties Window	53
1.3.5 TextDisplay Object	54
1.3.5.1 Overview	54
1.3.5.2 Interface	54
1.3.5.3 Property Pages	55
1.3.5.4 Properties Window	58



E-Prime Reference Guide
Table of Contents

1.3.6	ImageDisplay Object	60
1.3.6.1	Overview	60
1.3.6.2	Interface	60
1.3.6.3	Property Pages	62
1.3.6.4	Properties Window	63
1.3.7	Slide Object	65
1.3.7.1	Overview	65
1.3.7.2	Interface	65
1.3.7.3	Property Pages	67
1.3.7.4	Properties Window	68
1.3.7.5	SlideState Object	68
1.3.7.6	Sub-Objects	71
1.3.8	FeedbackDisplay Object	76
1.3.8.1	Overview	76
1.3.8.2	Interface	76
1.3.8.3	Property Pages	78
1.3.8.4	Properties Window	80
1.3.8.5	FeedbackDisplay Macros	82
1.3.9	InLine Object	83
1.3.9.1	Overview	83
1.3.9.2	Interface	83
1.3.9.3	Property Pages	84
1.3.9.4	Properties Window	84
1.3.10	SoundOut Object	84
1.3.10.1	Overview	84
1.3.10.2	Interface	84
1.3.10.3	Property Pages	85
1.3.10.4	Properties Window	86
1.3.11	Wait Object	88
1.3.11.1	Overview	88
1.3.11.2	Interface	88
1.3.11.3	Property Pages	88
1.3.11.4	Properties Window	88
1.3.12	Label Object	89
1.3.12.1	Overview	89
1.3.12.2	Interface	89
1.3.12.3	Property Pages	89
1.3.12.4	Properties Window	89
1.3.13	PackageCall Object	90
1.3.13.1	Overview	90
1.3.13.2	Interface	92
1.3.13.3	Property Pages	92
1.3.13.4	Properties Window	93
1.3.14	Scripting Objects	93
1.3.14.1	Canvas Object	93
1.3.14.2	Summation Object	93
1.3.14.3	Debug Object	93
<i>Chapter 2 : E-Basic</i>		95
2.1	Listing of E-Basic Commands	95
2.1.1	Audio Control	95
2.1.2	Compiler Control	95
2.1.3	Data Logging	95



E-Prime Reference Guide
Table of Contents

2.1.4	Data Type Conversion	96
2.1.5	Data Type Conversion: Casting	96
2.1.6	Date and Time	97
2.1.7	Display: Dialogs	97
2.1.8	Display: General Control	97
2.1.9	Display: Graphics	97
2.1.10	Error Handling	98
2.1.11	Experiment Control: Ports	98
2.1.12	Experiment Control: Stimulus	98
2.1.13	Experiment Control: Timing	98
2.1.14	File Input/Output	99
2.1.15	Input Device Control	99
2.1.16	Math	99
2.1.17	Math: Financial	99
2.1.18	Math: Randomization	99
2.1.19	Object Control	100
2.1.20	Operators	100
2.1.21	Program Control	100
2.1.22	String Manipulation	101
2.1.23	Variable Control	101
Chapter 3 : E-Run		103
3.1	About E-Run	103
3.2	About the Interface	103
3.2.1	Menus	104
3.2.1.1	File Menu	104
3.2.1.2	Edit Menu	104
3.2.1.3	View Menu	104
3.2.1.4	Build Menu	104
3.2.1.5	Help Menu	104
3.2.1.6	Context Menu	105
3.2.2	Toolbar	105
3.2.3	Script window	106
3.3	Using E-Run	106
3.3.1	Installation	106
3.3.1.1	Hardware Key	106
3.3.2	Compile versus Generate	107
3.3.3	Errors	107
3.3.3.1	Compile Errors	107
3.3.3.2	Run-Time Error	107
3.3.3.3	Locating Errors	108
3.3.4	Running Programs	109
3.3.5	Sharing Pre-Developed Programs	111
3.3.6	Create a Shortcut to an Experiment	111
3.3.7	Exiting E-Run	113
Chapter 4 : E-Merge		115
4.1	About the E-Merge Application	115
4.2	About the Interface	115
4.2.1	Menus	116
4.2.1.1	File Menu	116



E-Prime Reference Guide Table of Contents

4.2.1.2	Edit Menu	116
4.2.1.3	View Menu	116
4.2.1.4	Help Menu	117
4.2.1.5	Context Menu	117
4.2.2	Folder Tree View	117
4.2.3	File List View	118
4.2.3.1	Icons	118
4.2.3.2	File Name	118
4.2.3.3	Experiment	118
4.2.3.4	Status	119
4.2.3.5	Subject	119
4.2.3.6	Session	119
4.2.3.7	Last Merged	119
4.2.3.8	Last Modified	119
4.2.3.9	Created	120
4.2.3.10	Size in KB	120
4.2.4	Merge Log View	120
4.2.5	Status Bar	120
4.2.5.1	Status	120
4.2.5.2	Target File	121
4.2.5.3	Free Space	121
4.3	Using E-Merge	121
4.3.1	Annotations	121
4.3.2	Exiting E-Merge	121
4.3.3	Conflicts During Merge Operations	122
4.3.4	Filtering the File List View	125
4.3.5	Help	126
4.3.5.1	Contents	127
4.3.5.2	Index	127
4.3.5.3	Find	128
4.3.5.4	Context Sensitive Help	128
4.3.6	Icons	129
4.3.7	Merging Files	129
4.3.7.1	Selecting Source Files	129
4.3.7.2	Selecting Target Files	130
4.3.7.3	Merge Operations	130
4.3.7.4	Merge Results	135
4.3.7.5	Undo Merge	136
4.3.7.6	Merge Input History	137
4.3.7.7	Merge Output History	138
4.3.7.8	Security Levels	139
4.3.8	Moving Around in E-Merge	139
4.3.8.1	Folder Tree View	139
4.3.8.2	File List View	139
4.3.8.3	Merge Log View	140
4.3.8.4	Shortcuts	140
4.3.9	Opening E-Merge	140
4.3.10	Opening a Data File	141
4.3.11	Options	141
4.3.11.1	Initial Display Folder	141
4.3.11.2	Conflicts	141
4.3.12	Properties	142
4.3.12.1	General	142



E-Prime Reference Guide
Table of Contents

4.3.12.2	Contents	142
4.3.12.3	Merge Input History	143
4.3.12.4	Merge Output History	144
4.3.13	Quick Reference	144
Chapter 5 : E-DataAid		147
5.1	About the E-DataAid Application	147
5.2	About the Interface	147
5.2.1	Menus	148
5.2.1.1	File Menu	148
5.2.1.2	Edit Menu	148
5.2.1.3	View Menu	149
5.2.1.4	Tools Menu	149
5.2.1.5	Help Menu	149
5.2.1.6	Context Menu	150
5.2.2	Toolbar	150
5.2.3	Spreadsheet	150
5.2.3.1	Organization	151
5.2.3.2	Cells	152
5.2.4	Filters Bar	154
5.2.5	Status Bar	154
5.3	Using E-DataAid	154
5.3.1	Analyzing	154
5.3.1.1	Creating an Analysis	155
5.3.1.2	Binning Data	157
5.3.1.3	Running an Analysis	158
5.3.1.4	Saving an Analysis	165
5.3.1.5	Loading an Analysis	166
5.3.1.6	Creating a New Analysis	166
5.3.1.7	Closing the Analyze dialog	166
5.3.1.8	Batch Analyses	166
5.3.2	Annotations	168
5.3.2.1	Displaying Annotations	168
5.3.2.2	Adding Text to Annotations	169
5.3.2.3	Creating New Annotations	170
5.3.2.4	Viewing Details	171
5.3.2.5	Exporting Annotations to Text	171
5.3.3	Arranging the Spreadsheet	172
5.3.3.1	Spreadsheet Organization	172
5.3.3.2	Selecting Columns	173
5.3.3.3	Resizing Columns	173
5.3.3.4	Arranging Columns	173
5.3.3.5	Hiding/Unhiding Columns	175
5.3.3.6	Restoring Column Order	176
5.3.3.7	Collapsing Levels	176
5.3.3.8	Restoring Spreadsheet	177
5.3.4	Closing a Data File	177
5.3.5	Editing Data	178
5.3.5.1	Editing Experiment Name	179
5.3.5.2	Edit Subject and Session Number	179
5.3.5.3	Editing Cell Values	180
5.3.5.4	Adding Variables	180



E-Prime Reference Guide
Table of Contents

5.3.5.5	Renaming Variables	181
5.3.5.6	Copying Data	182
5.3.5.7	Pasting Data	182
5.3.5.8	Fill Down	182
5.3.5.9	Deleting Data	182
5.3.5.10	Undoing Edits	183
5.3.5.11	Finding Data	183
5.3.5.12	Replacing Data	184
5.3.6	Exiting E-DataAid	185
5.3.7	Exporting	185
5.3.7.1	Export Data to Excel	185
5.3.7.2	Export Data to StatView	186
5.3.7.3	Export Data to E-Prime Text	187
5.3.7.4	Export Data to Other Format	188
5.3.8	Filtering	189
5.3.8.1	Checklist Filter	189
5.3.8.2	Range Filter	190
5.3.8.3	Display of Active Filters	192
5.3.8.4	Clearing Filters	193
5.3.8.5	Refreshing Filters	194
5.3.9	Help	194
5.3.9.1	Contents	194
5.3.9.2	Index	194
5.3.9.3	Find	195
5.3.9.4	Context Sensitive Help	196
5.3.10	Importing Data Files	196
5.3.10.1	MEL Professional Data File	196
5.3.10.2	PsyScope Data File	197
5.3.10.3	E-Prime Text File	199
5.3.11	Opening a File	200
5.3.11.1	Opening as User	200
5.3.11.2	Opening as Administrator	201
5.3.12	Options	202
5.3.12.1	General	202
5.3.12.2	Spreadsheet	203
5.3.12.3	Edits	204
5.3.12.4	Filters	204
5.3.13	Printing	205
5.3.13.1	Print Setup	205
5.3.13.2	Print Preview	206
5.3.13.3	Print	207
5.3.14	Playing a WAV file	207
5.3.15	Properties	207
5.3.15.1	General	208
5.3.15.2	Contents	209
5.3.15.3	Columns	209
5.3.16	Saving a Data File	210
5.3.17	Security	210
5.3.17.1	Opening a File as an Administrator	210
5.3.17.2	Variable settings	210
5.3.17.3	General Settings	211
5.3.17.4	Annotation Settings	211



6.1	About E-Recovery	213
6.2	About the Interface	213
6.2.1	The Progress Dialog Options	214
6.2.1.1	Full	214
6.2.1.2	Simple	214
6.2.1.3	None	215
6.2.2	Command Buttons	215
6.2.2.1	About E-Recovery	215
6.3	Using E-Recovery	215
6.3.1	Recovering Files	216
6.3.2	Viewing Recovered Data	216
6.3.3	Troubleshooting – Common Data Conversion Problems	217
6.3.3.1	Data File Open	218
6.3.3.2	Reserved Words	218
6.3.3.3	Logging Turned Off	218
6.3.3.4	Extra Tab After an Attribute	218
6.3.4	Exiting E-Recovery	218
6.3.5	Getting Help	219
<i>Appendix A - Error List</i>		A-1



Chapter 1: E-Studio

1.1 About the E-Studio Application

The E-Studio application is used to design experiments. Experiments are created graphically by dragging and dropping objects onto procedural timelines, and then setting the object's properties. E-Studio features standard interface characteristics including menus, toolbars and status bars. E-Studio follows a single document model, meaning that only one experiment can be open at a time. Within a single experiment, multiple windows will typically be open in E-Studio simultaneously.

1.1.1 The Menus

Like most applications, E-Studio features a variety of menus. The menu bar is located at the top of the application window.



Menus are lists of related commands. For instance, an Edit menu typically contains commands allowing the user to Cut, Copy and Paste items; commands performing similar duties are grouped together within the same menu.

Commands are selected by clicking on them with the mouse. Alternatively, *accelerator and short-cut keys* are provided. These short-cuts, sometimes referred to as "Hot keys", are key combinations typically including the Ctrl, Shift or Alt keys pressed simultaneously with another key. They activate the corresponding command as if selected from a menu. It is not necessary to open a menu to activate the hot key selection method.

An ellipse (...) following a command in a menu signifies that a *dialog box* will open when that command is selected. The dialog box contains the necessary option information for completing that command. Some menu commands contain sub-menus. The sub-menus function just like regular menus (i.e., commands are selected by clicking on them with the mouse).

Shortcut menus are also available for many objects. Also called *context menus*, they are a collection of related commands pertaining to a specific object. Access these menus by right clicking an object.

1.1.1.1 E-Studio Menus

An ellipse (...) following a command in a menu signifies that a *dialog box* will open when that command is selected. The dialog box contains the necessary option information for completing that command. The Options command makes use of the Property pages to set specific global options for the E-Studio application.



File Menu

Command	Shortcut	Function
<u>New...</u>	Ctrl+N	Creates a new E-Studio Experiment file.
<u>Open...</u>	Ctrl+O	Displays a dialog box to open an existing *.ES file.
<u>Save</u>	Ctrl+S	Saves the current *.ES file.
<u>Save As...</u>	F12	Displays a dialog box to save the current *.ES file under a new name.
<u>Send...</u>		Launches the e-mail client to send the current file to an e-mail address.
(List of recently opened *.ES files)		
<u>Exit</u>	Alt+F4	Exits the E-Studio application.

Edit Menu

Command	Shortcut	Function
<u>Undo</u>	Ctrl+Z	Reverses the last action performed (when possible).
<u>Redo</u>	Ctrl+Y	Repeats the last action performed (when possible).
<u>Cut</u>	Ctrl+X	Removes the selected text and places it on the clipboard.
<u>Copy</u>	Ctrl+C	Places a copy of the selected text on the clipboard.
<u>Paste</u>	Ctrl+V	Inserts the contents of the clipboard in the current location of the cursor.
<u>Rename</u>		Allows the selected object to be renamed. This alters the Name property of the object. Refer to section 1.3 for naming restrictions.
<u>Delete</u>	DEL	Deletes the object from its selected location.
<u>Experiment...</u>	Ctrl+E	Opens the Property pages for the Experiment object.
<u>Properties...</u>	Alt+Enter	Permits the setting of properties for the Structure window.



View Menu

The View menu contains a list of various windows, which can be viewed within E-Studio. The list is comprised of toggle options, indicating whether or not a specific window is displayed. When a window is currently in view, a checkmark appears to the left within the View menu.

Window	Shortcut	Function
<u>Attributes</u>	Alt+1	Reveals a list of all attributes declared in the current experiment.
<u>Browser</u>	Alt+2	Reveals a window containing all objects (referenced and unreferenced) in the current experiment.
<u>Output</u>	Alt+3	Reveals a window displaying details of the status of performed operations.
<u>Properties</u>	Alt+4	Reveals a window displaying the property settings for the currently selected object.
<u>Script</u>	Alt+5	Opens a window in the application area containing both user written script, and the complete script generated by E-Studio.
<u>Structure</u>	Alt+6	Reveals a window displaying the hierarchical structure of the objects in the experiment.
<u>Toolbox</u>	Alt+7	Reveals the toolbar containing the E-Objects available for experiment creation.
<u>Full Screen</u>		Displays the Workspace as full screen.
<u>Toolbar</u>		Reveals the toolbar at the top of the application window just below the menus.
<u>Status Bar</u>		Reveals the Status bar along the bottom of the application window.

E-Run Menu

Command	Shortcut	Function
<u>Generate</u>	Ctrl+F7	Generates the E-Basic script based on the current settings in E-Studio. By default, the experiment is saved and compiled each time the script is generated. These options may be changed through the Options command in the Tools menu.
<u>Run</u>	F7	Generates, compiles, and executes the script.

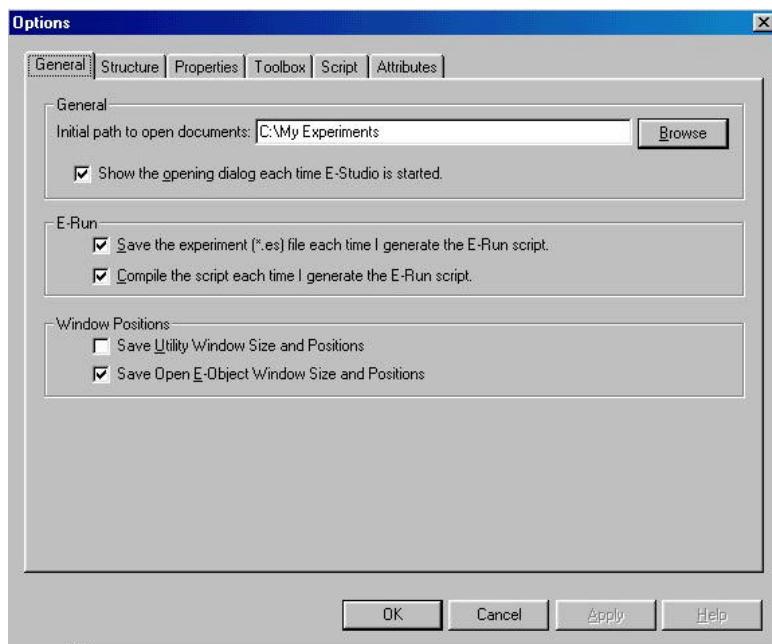
Tools Menu

Command	Shortcut	Function
<u>E-Run</u>		Launches the E-Run application.
<u>E-DataAid</u>		Launches the E-DataAid application.
<u>E-Merge</u>		Launches the E-Merge application.
<u>Reset Utility Windows</u>		Resets utility windows sizes and positions to the defaults.
<u>Options...</u>		Specify the global options for the E-Studio application.



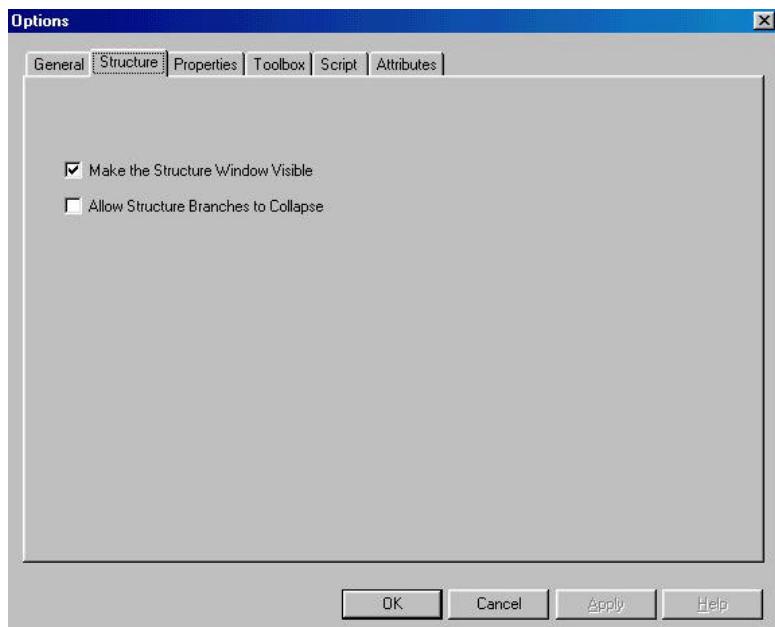
General Tab

The General tab allows the setting of the initial path for the opening of documents. This tab also sets specific properties for E-Run in regards to compiling and generating script, and the window positions in E-Studio.



Structure Tab

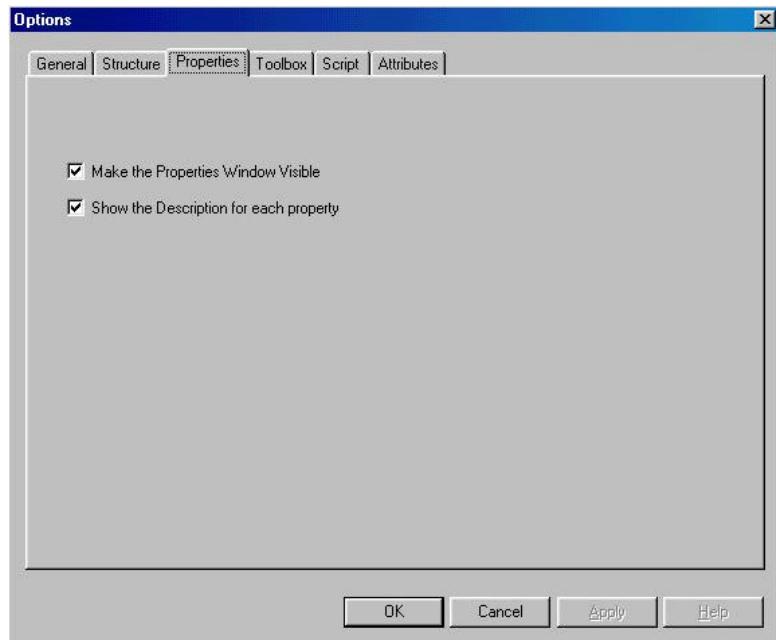
The Structure tab allows the user to set the parameters concerning the Structure window, and to control the collapsing of branches within the Structure view.





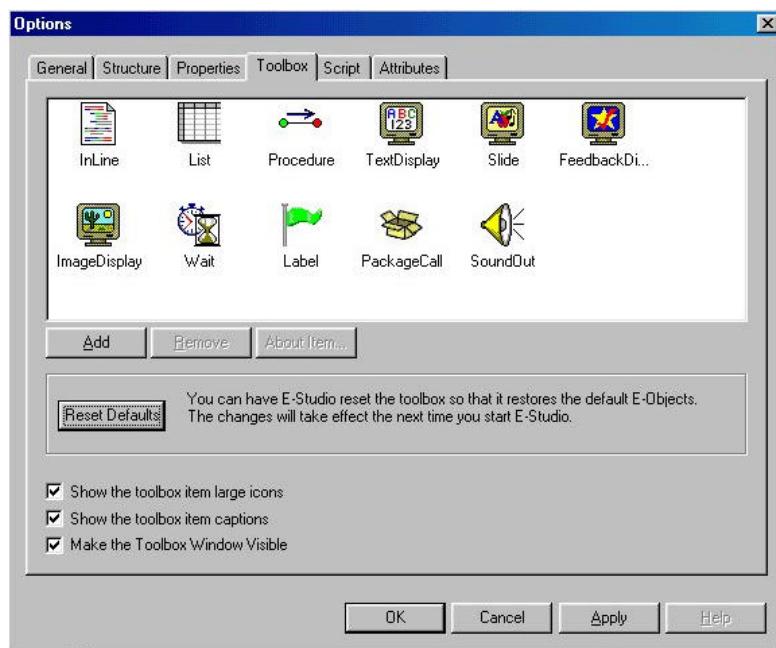
Properties Tab

The Properties tab allows the user to make the Properties window visible along with the property descriptions.



Toolbox Tab

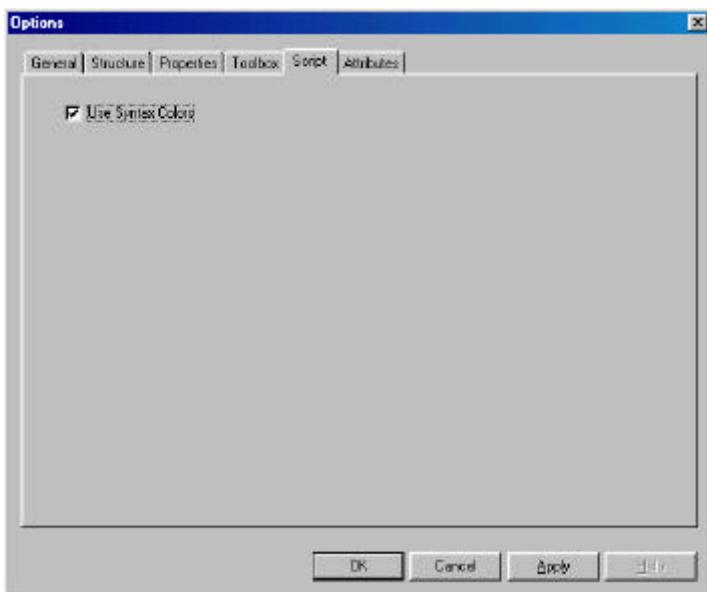
The Toolbox tab allows the user to set specific properties regarding the Toolbox. By default, all currently active E-Objects are listed.





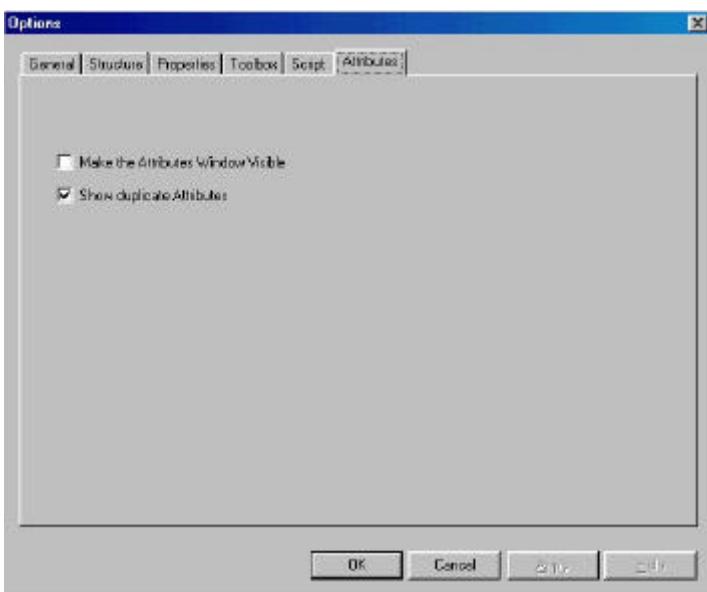
Script Tab

The Script tab allows the user to set whether or not syntax highlighting is used within the script. These colors are applied to user script as well as the full script. Syntax highlighting is discussed in section 1.2.6 of the current chapter.



Attributes Tab

The Attributes tab allows the user to make the Attributes window visible each time E-Studio is opened, and also grants the user the ability to show duplicate Attributes. Duplicate attributes occur when multiple List objects define attributes with identical names. If duplicate attributes exist, the Attributes window will show the name of the attribute multiple times paired with the names of the Lists containing the duplicate. If Show duplicate Attributes is unchecked, the union of all attributes will be displayed.





Window Menu

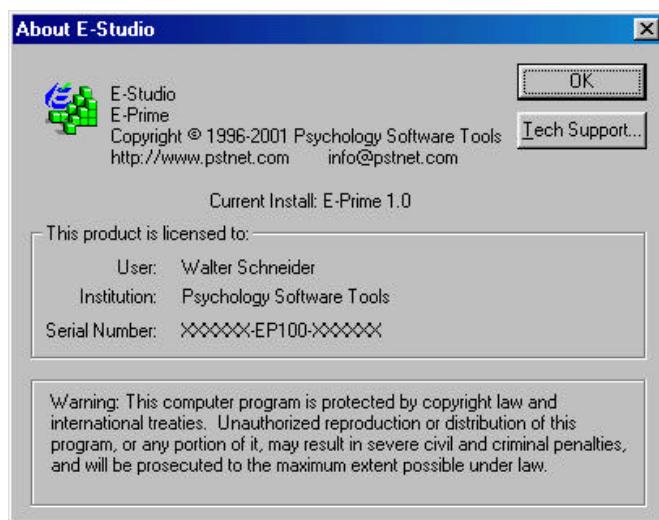
Command	Function
<u>Cascade</u>	Arranges windows open in the Workspace overlapping in a cascade formation.
<u>Tile Horizontally</u>	Arranges windows open in the Workspace in rows.
<u>Tile Vertically</u>	Arranges windows open in the Workspace in columns.
<u>Close All</u>	Closes all windows open in the Workspace.

Help Menu

Command	Function
<u>Help Topics</u>	Opens the on-line Help for E-Studio.
<u>PST on the Web...</u>	Launches the default web browser and connects to the PST web site.
<u>E-Basic Help</u>	Opens the on-line help for the E-Basic scripting language.
<u>About E-Studio...</u>	Opens a dialog box revealing the application information, including version number and serial number.

About E-Studio

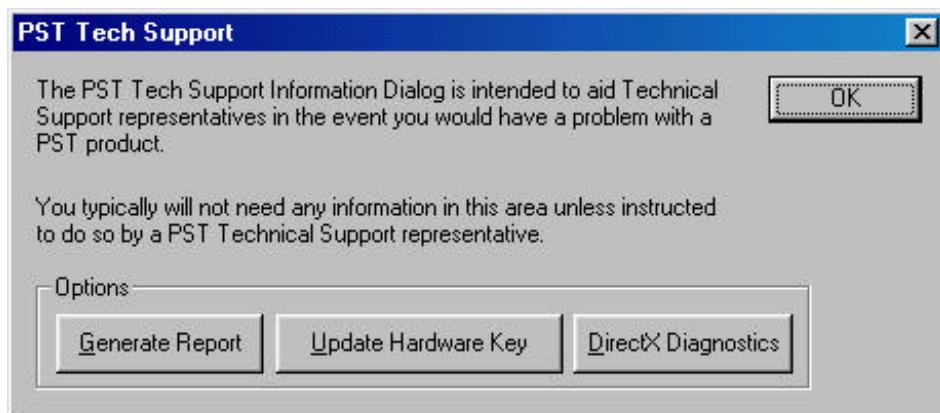
When contacting PST for technical support, the user must provide a valid serial number, which can be found here in the About E-Studio dialog box. In addition to providing the user with the application and product licensing information, the About E-Studio dialog box also includes a helpful tool used by PST technical consultants to collect information regarding the user's machine and E-Prime.



Clicking the Tech Support button displays the PST Tech Support dialog, listing the available options to help diagnose problems that may occur within the system. This will help to determine causes of problems in relation to the machine and version of E-Prime.



The PST Tech Support dialog box will appear which includes three options; Generate a Report, Update the Hardware Key or run DirectX Diagnostics. A PST Technical Support representative will instruct which option to choose in order to help remedy a problem that may be occurring. As noted in the dialog box, users need only select from the following options if instructed to do so by a PST Technical Support representative.



1.2 About the Interface

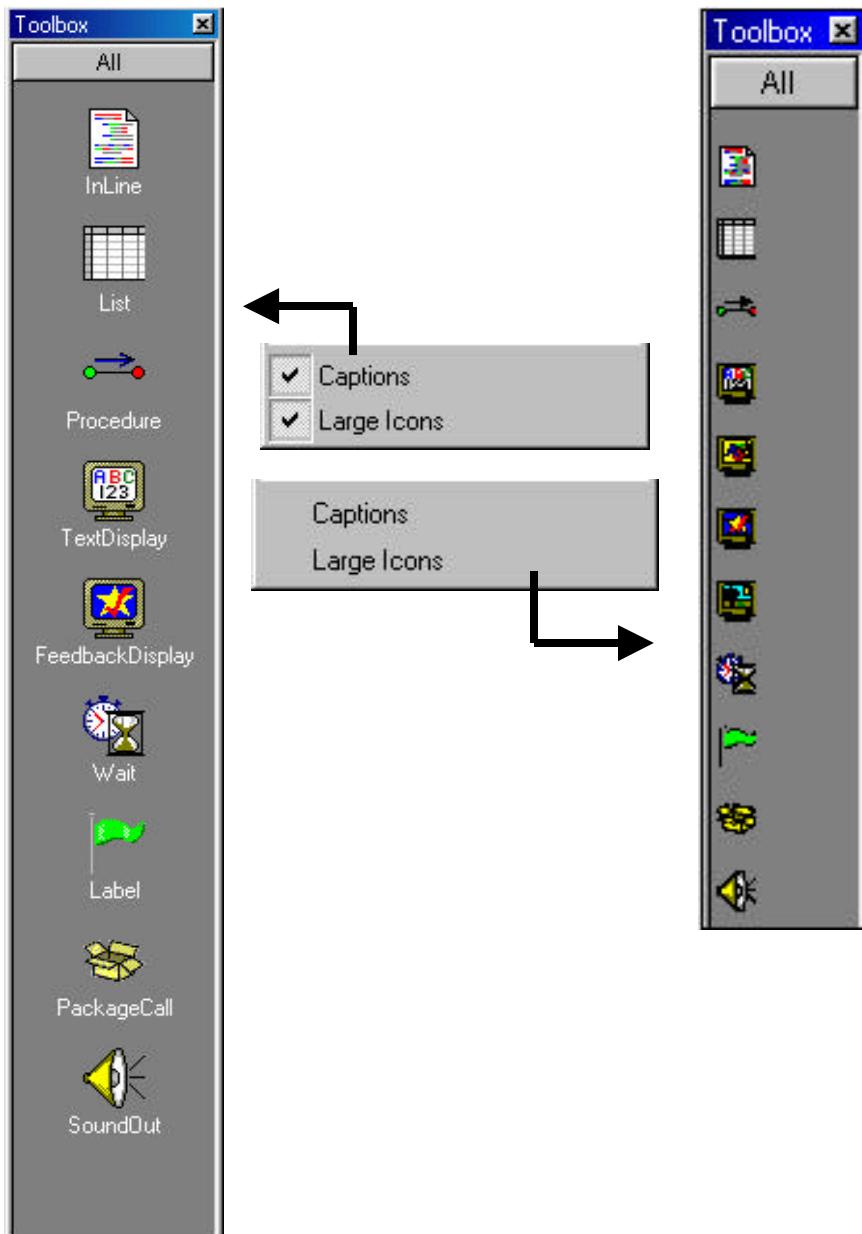
In addition to the standard interface features, E-Studio features the following specific window views:

- Toolbox
- Structure view
- Workspace
- Browser
- Properties window
- Attributes window
- Script window

Each of these window views can have only one instance and is docked to the sides of the application window by default. The exception to this is the Workspace. The Workspace is the large area of the application in which window representations of objects are opened. Each of the window views will be discussed individually in the next section.

1.2.1 Toolbox

The *Toolbox* is a framed window that displays the icons of all currently registered E-Objects. The *Toolbox* is docked by default along the left side of the application window, and can be resized using standard platform sizing and positioning mechanisms. The E-Objects within the *Toolbox* can be customized via the context menu to be displayed using either large or small icons, and with or without descriptive labels (i.e., captions). Right click to display the context menu.

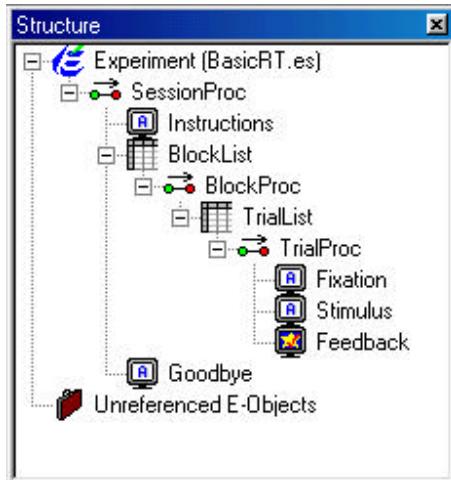


The Toolbox context menu may be accessed by right-clicking within the Toolbox window. The Toolbox can be removed from the visible application window by clicking the small X in the upper right corner of the window or by removing the checkmark next to the Toolbox option in the View menu. The Toolbox may be redisplayed via the View menu (i.e., select the Toolbox option in the View menu). Visual feedback (i.e., outline) is provided when the cursor is placed over the E-Objects to indicate which object is selected.



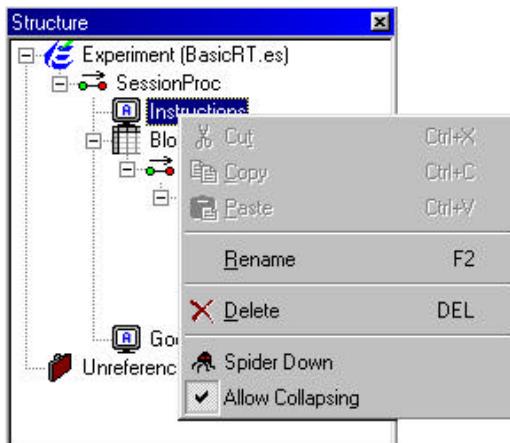
1.2.2 Structure View

The purpose of the *Structure view* is to provide a hierarchical representation of the events within the current experiment. The Structure view is displayed as a framed window docked, by default, to the immediate right of the Toolbox, and features a collapsible tree view of the experiment structure.



The Structure view enables the user to easily review and modify the outline of a developing experiment, and to visually inspect the overall experiment based on the placement of the objects. Both objects that are currently in use (e.g., referenced by a Procedure), and objects that have been created but are not directly referenced are visible within the tree. The latter are located in the Unreferenced E-Objects section of the tree. Objects not explicitly referenced (i.e., either unused, or called via script but not called by a Procedure) are listed in the Unreferenced E-Objects folder.

While designing an experiment, much of the user's time will be spent working in, or in conjunction with, the Structure view. New objects may be created by dragging them to a specific location within the tree in the Structure view. Furthermore, objects may be repositioned within the tree by dragging them to the desired location. Each object located within the tree features a context menu (accessed by right clicking on the object), which permits standard editing features (e.g., Rename, Delete).





Double clicking on an object within the Structure view results in the window representation of that object being opened within the Workspace. When an object window is open in the Workspace, any reorganization of that object via the Structure view (e.g., adding or moving objects in a Procedure) will be echoed in the object's window in the Workspace.

1.2.2.1 Collapsible Tree View

The Tree view in the Structure view is collapsible, to facilitate the viewing of specific branches within the tree. The collapsible tree features small boxes to the left of *parent objects* (i.e., objects containing other objects). When a branch of the tree is collapsed, the small box to the left of the parent object will contain a plus (+) sign. To expand a parent object branch, click on the small box containing the plus sign. When a branch is expanded, the small box to the left of the parent object will contain a minus (-) sign. To collapse a parent object branch, click in the small box containing the minus sign. To expand an entire branch of the tree, including all sub-branches, select the parent object, and right click on the object to reveal its context menu. Choose the *Spider Down* option to expand the selected branch, and all of its sub-branches. Alternatively, the tree can be set to always expand by unchecking the Allow Collapsing option in the context menu.



1.2.2.2 Renaming Objects

An object may be renamed in the Structure view by selecting any instance of the object in the tree and then clicking it again to get an in-place edit box, by pressing {F2} to enter edit mode, or by right-clicking to display the context menu and selecting the Rename command. Refer to section 1.3 (this chapter) for restrictions on naming objects.

1.2.2.3 Deleting Objects

A reference to an object may be deleted in the Structure view by selecting the object and pressing the Del key, selecting the Delete command from the Edit menu, or by selecting the Delete command from the context menu. Objects within the Structure view are actually only references or short-cuts to the object, rather than the object itself. When an object is deleted from a Procedure in the structure of the experiment, only the specific reference to the object is being deleted. Thus, it is possible to delete one reference to the object without affecting other references to the same object.

If all references to the object are deleted (i.e., the object is no longer referenced by another object within the structure), the object is moved to the Unreferenced E-Objects section of the tree. The purpose of this intermediate step is to allow users to delete objects from specific Procedures while allowing them to be used by others, or to remain in Unreferenced E-Objects for use at a later time. If the delete operation is requested for an object located in the Unreferenced E-Objects section or the Browser, the user is notified that this action will remove all references to the object and it will be destroyed forever, and the user is asked to confirm the operation.



1.2.2.4 Moving Objects

Objects may be repositioned within the Structure view tree by clicking and dragging them to the desired location. Objects may also be moved to a new location within a Procedure window opened in the Workspace by clicking the object and dragging it to the new location. To move objects from one Procedure to another, click the object to select it. Then, while holding the shift key down, click and drag the object to the new Procedure. Any object movement within the Workspace will be reflected in the Structure view.

To aid in the positioning of objects, the arrow cursor () is displayed to indicate the placement of the object within the structure. Some moves are illegal (e.g., a Procedure cannot be placed on another Procedure). In the case of illegal moves, the cursor will change to , indicating that the move cannot be completed.

Moves within a Procedure will reposition the object reference. Moves between Procedures will create new references to the same object, thus allowing a single object to be referenced within multiple Procedures without making duplicate objects.

1.2.2.5 Copying Objects

Exact copies of objects may be created within the Structure view tree by clicking on the object to select it, then while holding down the Ctrl key, dragging the object to the desired location. Each new object created will be assigned a default name of the format "ObjectNameN, where "N" is an automatically assigned integer (i.e., Fixation, Fixation1, Fixation2). Objects may also be copied within a Procedure window opened in the Workspace by applying the same technique. Any copying of objects within the Workspace will be reflected in the Structure view. Finally, copies of objects may be created in the Browser window by right-clicking on an object to display the context menu and selecting the Copy command. Redisplay the context menu to paste the copy into the Browser window.

1.2.2.6 Opening Objects

An existing object's window representation may be opened in the Workspace by double clicking the object in the Structure view. To open a new instance of an object, click and drag the object icon from the Toolbox to the Workspace, or to a specific location within the Structure view. Visual cues (arrows) are presented to indicate where an object will be placed when the mouse button is released. If a new object is created in the Workspace, that object will be placed in the Unreferenced E-Objects folder in the Structure view. When dragging a new object to a parent object that is collapsed in the Structure view, hovering over the parent object will open the window representation for the parent object in the Workspace.

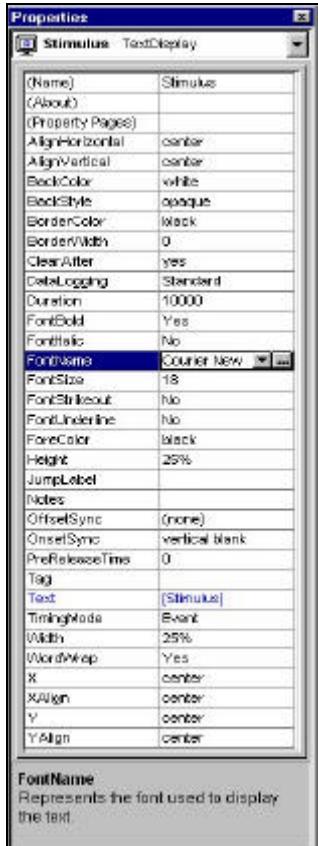
1.2.3 Properties Window

The Properties window is a dockable window used to display all of the standard properties and property values associated with objects used in the experiment. By default, the Properties window appears to the immediate right of the Toolbox, just below the Structure view. The Properties window can be a free-floating window, or can be docked to any side of the application window. The Properties window appears in the E-Studio application as a two-column list/grid. This window is designed to be virtually identical in appearance/function to the one used by VisualBasic™ and other common development environments.

When an object is selected within E-Studio, the Properties window is updated to display the properties associated with the selected object. The properties are listed in alphabetical order



within the Properties window with the exception of commonly used properties, which appear at the top of the list.



Properties may be edited within the Properties window by clicking on the property and typing the desired value, or by choosing an option from the dropdown box (when available). When a property is selected within the Properties window, a brief description of that property appears in the bottom of the window. The display of the property description may be toggled on or off by right clicking in the Properties window in the area below the properties, and selecting or deselecting the Description option from the context menu.

Properties may also be set as variable information (i.e., attribute references). Property values may be set to vary at run-time according to the values of an attribute. The name of the attribute to reference is enclosed within square brackets (e.g., [attribute]). This syntax indicates to E-Studio that the value for the property field will be resolved at run-time by referring to the named attribute to obtain its current value.

1.2.4 Attributes

The *Attributes window*, not visible by default, is a dockable window. When the Attributes window is opened via the View menu, it is located on the right side of the application window by default. It features a list of every user-declared attribute within the experiment's List objects, regardless of the List in which it resides.



The screenshot shows three windows in E-Studio:

- BlockList**: Summary: 1 Sample (1 cycle x 1 sample/cycle). 1 Cycle equals 1 sample. Sequential Selection. Table:

ID	Weight	Procedure	Nested	PracticeMode
1		1 BlockProc		No
- TrialList**: Summary: 4 Samples (1 cycle x 4 samples/cycle). 1 Cycle equals 4 samples. Random Selection. Table:

ID	Weight	Procedure	Nested	Stimulus	CorrectAnswer
1	2	2 TrialProc		X	1
2	2	2 TrialProc		Y	2
- Attributes**: Name:
 - CorrectAnswer
 - PracticeMode
 - Stimulus

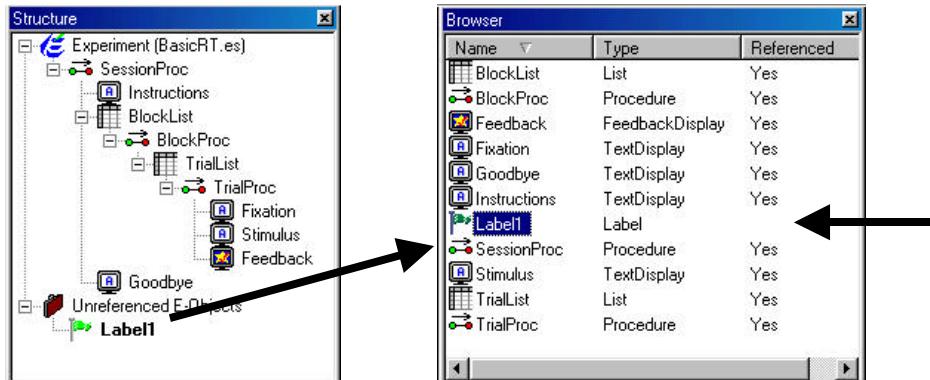
Arrows point from the "PracticeMode" column in the BlockList table and the "CorrectAnswer" column in the TrialList table to the "Attributes" window, indicating that these attributes can be dragged and dropped into the respective columns.

Attributes are typically the major independent and control variables used within the experiment (e.g., condition, stimulus, correct answer). The list of attributes can be sorted in ascending or descending alphabetical order by clicking on the header. Subsequent clicks on the header reverse the order of the sorting (e.g., A to Z, Z to A).

Attributes listed in the Attributes window can be dragged and dropped to various locations within the experiment (e.g., List object cells, Properties window, fields within object Property pages). When an attribute is dragged and dropped into a cell either in the Properties window or the List object, the attribute name appears enclosed in square brackets, and is displayed in blue. This syntax indicates to E-Studio that the value is variable, and will be resolved at run-time by referring to the named attribute to obtain its current value.

1.2.5 Browser

The *Browser* is a dockable window displaying the objects currently instantiated in the experiment in a report view format. The *Browser* is not visible by default, and may be displayed using the View menu. The report lists the name of the object, its type, and whether or not it is currently referenced. The objects listed are not necessarily used in the experiment, but they exist and are available for use. For example, an object that has been deleted from all Procedures, but not from the experiment, would appear in the Unreferenced E-Objects folder in the Structure view, and would be listed in the *Browser*. The value for the Referenced column in the *Browser* window would remain blank.



1.2.5.1 Sorting Objects

The objects listed in the Browser can be sorted by Name, Type, or Referenced status by clicking on the column header. Subsequent clicks on the header reverse the order of the sorting (e.g., A to Z, Z to A).

1.2.5.2 Deleting Objects

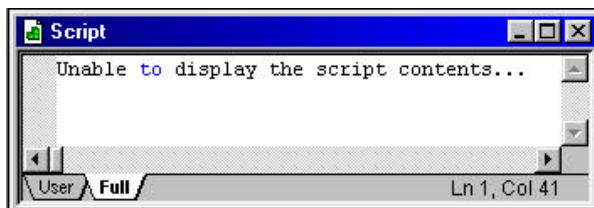
Deleting an object from the Browser results in all references to the object being removed from the experiment. Once deleted from the Browser, the object is not retrievable, and must be recreated if needed. A delete operation within the Browser requires confirmation from the user before it is performed.

1.2.5.3 Cut, Copy, and Paste

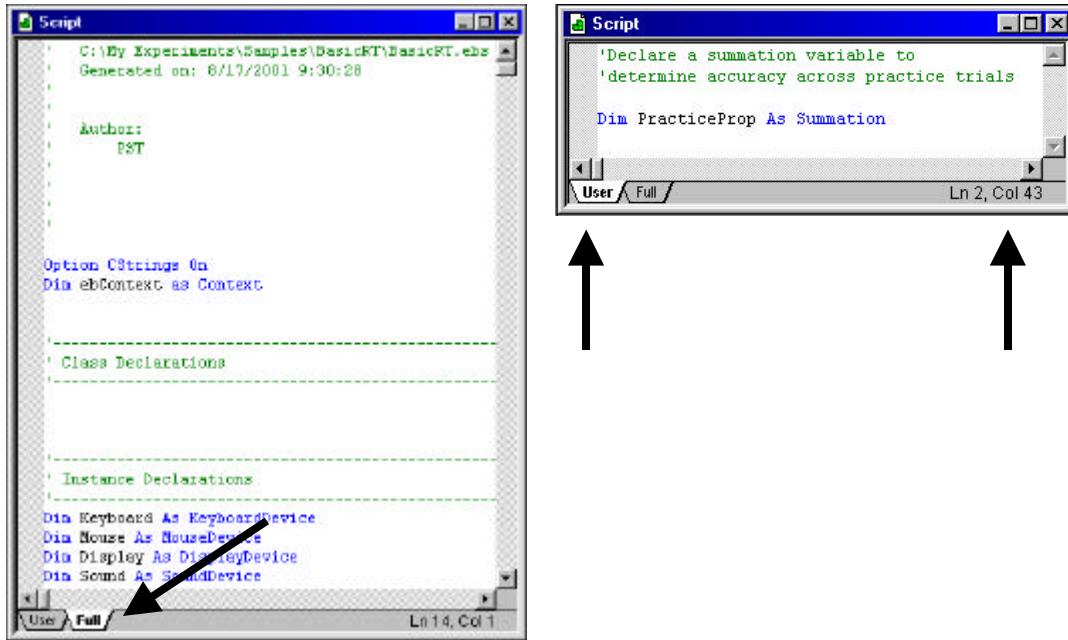
The Browser supports the dragging of objects from the Browser to other areas of the experiment. By default, a simple drag creates a new “reference” to the selected object, but a Ctrl key can be applied, as a mouse modifier, to indicate that a copy of the current item should be made. Specifically, Ctrl+Copy is often used to duplicate an object (e.g., to create a Procedure with the same sequence of events, but which calls a different List object). Likewise, cut, copy and paste options are available via the context menu, which can be accessed by right clicking on any object within the Browser.

1.2.6 Script Window

The Script window features two tabs, User and Full. Until the experiment is generated, the Full tab indicates that the script contents cannot be displayed.

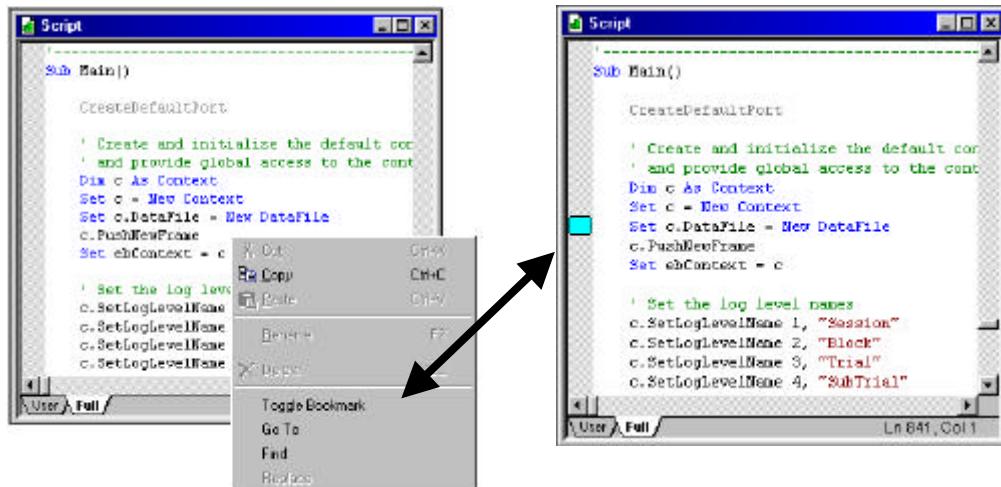


Once generated, the Full tab displays the entire script as generated based on the object placement and specification within E-Studio. The User tab enables user-defined subroutines or global variable declarations.



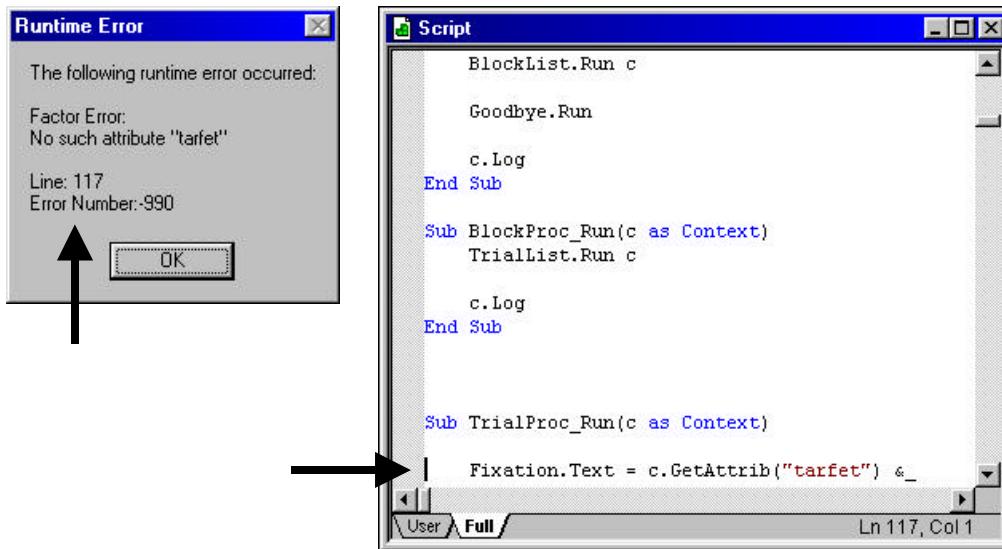
The Script window features syntax highlighting similar to that in the Visual Basic™ environment. Comments are featured in green, keywords in blue, statements in black, constants in orange, API in gray, and strings in burgundy. The Ln/Col data in the bottom right corner of the window indicates the current position of the cursor. The full script may not be edited (i.e., the Full tab is read-only).

Both tabs in the Script window feature a bookmark option via the context menu (right click inside the Script window). By selecting the "Toggle Bookmark" command in the context menu, a blue rectangle is inserted to the left of the line where the cursor is located. This is a useful feature for locating specific lines in the script for debugging purposes. To remove a bookmark, place the cursor on the line next to the bookmark and select Toggle Bookmark again from the context menu. Bookmarks may also be toggled using **Ctrl+{F2}**. Use the {F2} function key to jump between bookmarks.



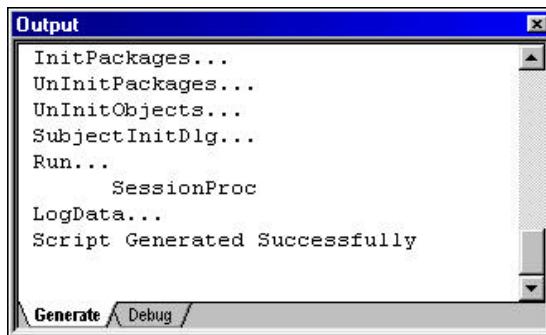


By default, the Script window is not open when E-Studio is opened, but may be displayed using the View menu. The Script window automatically opens when a script or compile error is generated. When an error occurs, the cursor is placed in the full script at the location at which the error occurred.



1.2.7 Output Window

The *Output window* is a two-tabbed window appearing at the bottom of the E-Studio display window by default.



The Generate tab provides feedback to the user concerning the generation of the E-Basic script. The Debug tab displays any error messages generated during run-time, or debugging information generated through the use of the Debug.Print command in an InLine object (refer to section 1.3.14-*Scripting Objects* in this chapter and 4.4.6-*Debugging* in the User's Guide Chapter 4-*Using E-Basic*).

A context menu may be displayed by right-clicking in the Output window. The context menu allows the user to copy the contents of the Output window to the clipboard for pasting into another application (e.g., Word or Excel), to clear the window of the current contents, or to search within the output text for a key phrase.



1.3 E-Objects

E-Objects are design time building blocks of an experiment, and are located in the Toolbox within E-Studio. The E-Objects in the Toolbox are used to build an experiment by dragging and dropping the icons onto valid targets. Valid targets include the Structure view, procedural timelines, or the Workspace. When an object is dropped into the Structure view or on a procedural timeline, the placement of the object is specific to where it was dropped (e.g., on the trial Procedure). Visual cues (i.e., arrows) concerning the placement of the object are given when dropping an object within the Structure view or onto a Procedure window opened in the Workspace. When a new object is created by dragging an icon to the Workspace, a new instance of that object is created. The reference to this object is placed in the Unreferenced E-Objects folder in the Structure view.

Each object created within an experiment has a set of associated properties. The associated properties can be customized and specified either in the Properties window or in the object's Property pages. Once part of an experiment, an object's properties may be viewed by highlighting it (i.e., by selecting it) in one of the window representations of the experiment (e.g., Structure view, Browser, Workspace).

E-Object	Icon	Category	Purpose
Experiment		Flow Control	Control highest-level information for the experiment including initialization of context variables, startup information, device properties, etc.
Procedure		Flow Control	Represents the sequential execution of the specified objects.
InLine		Flow Control	Allows user written script to be inserted at a specific point during program execution.
List		Sampling	A spreadsheet representation of attributes which can be sampled in a specified manner.
TextDisplay		Stimulus Presentation	Method to present a text string or character in a specified font, size and color.
ImageDisplay		Stimulus Presentation	Method to present an image.
Slide		Stimulus Presentation	Method to display combinations of text, images, and sound.
Wait		Timing and Synchronization	Method used to wait for a specified time.
SoundOut		Multimedia Presentation	Method to play WAV files.
FeedbackDisplay		Miscellaneous	Method to display customized feedback based on a response collected.
Label		Flow Control	Method to allow program execution to jump to a specific location within a Procedure.
Package		Flow Control	Method to call a block of E-Basic script in an external file.



Each new instance of an object is assigned a default name of the format "ObjectTypeN", where "N" is an automatically assigned integer that helps to create unique names. Although an object may be used in multiple instances throughout an experiment (i.e., the same object may be referenced in various locations within the structure of the experiment) all unique objects in the experiment must feature unique names. Object names cannot contain spaces and must observe the same rules for defining legal variable names in E-Basic (e.g., the following characters are not permitted: space /*&_^(){}\$#@%). Restrictions for object names are listed in the table below.

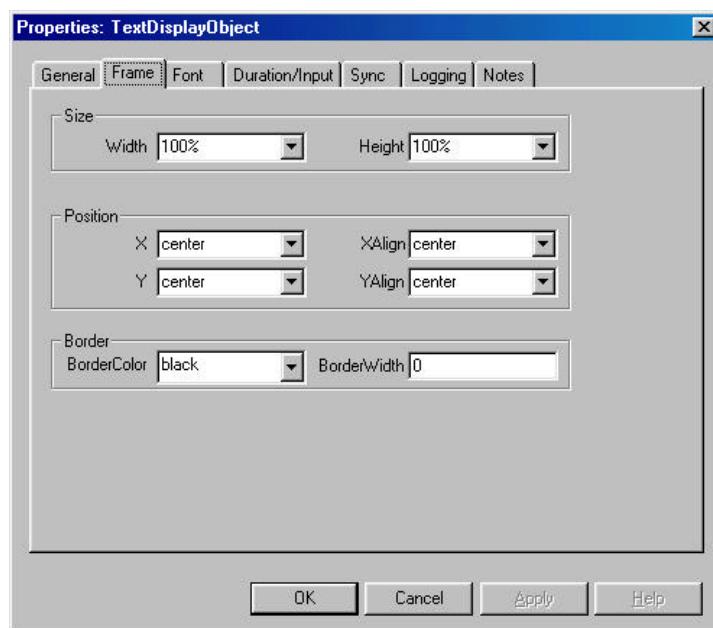
Rules for naming objects
Cannot be empty
Must begin with an alpha character
Permitted characters include a-z, A-Z, 0-9
Cannot exceed 80 characters
Cannot conflict with device name (e.g., Keyboard) or device type (e.g., KeyboardDevice)
Cannot be the same as any log level name (refer to Data File tab in Experiment object)
Cannot be the same as any type in the Toolbox (e.g., TextDisplay, Wait, etc.)
Cannot be named "c" (this character is reserved for the Context object)

1.3.1 Standard Property pages

Each of the E-Objects includes Property pages along with a Properties window. The Property pages may be displayed by clicking the Property Pages button when the object is opened in the Workspace, or via the Properties window. The advantage of the Property pages is the user-friendly interface which groups the associated properties into common categories represented as tabs. Five tabs are common to most E-Objects: Frame, Duration/Input, Sync, Logging, and Notes. These tabs are discussed in this section. Tabs that are unique to individual E-Objects are discussed in the Property pages sections for the specific objects

1.3.1.1 Frame Tab

The Frame tab is used by visual display objects to define a rectangular area on the screen as the display area. The rectangular area defined by the coordinates is referred to as a "Frame."





The Size fields set the size of the frame either in pixels, or as a percentage of the entire display area. In most cases, the display area refers to the entire screen. However, in the case of sub-objects used to compose SlideState objects on the Slide object, the display area refers to the Frame defined by the SlideState. For example, if the SlideState object sets the width of the Frame to 50%, the 50% setting refers to the screen size (e.g., in an 800x600 resolution, the 50% Frame width would translate to 400 pixels). If a sub-object on that SlideState sets the Frame width to 50%, this setting refers to the display area defined by the SlideState object, or 50% of the 400 pixels, translating to 200 pixels, or 25% of the entire screen. The values for the Width and Height fields may be chosen from the available options in the dropdown boxes, or a value may be entered directly. To indicate the value as a percentage of the entire display area, include the "%" symbol after the value. Otherwise, the values will be assumed to be pixel coordinates.

Field	Possible Values	Description
Width	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Sets the width of the display area to a percentage of the x (horizontal) resolution, or to a fixed pixel size.
Height	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Sets the height of the display area to a percentage of the y (vertical) resolution, or to a fixed pixel size.

The Position fields (X, Y, XAlign, YAlign) allow the positioning of the defined frame in the display area in relation to an anchor point. This allows the user to define areas on the display to aid in the placement of stimuli. The X and Y coordinates may be entered directly, or chosen from the available options in the dropdown boxes. To indicate the value as a percentage of the entire display area, include the "%" symbol after the value. Otherwise, any numeric values entered will be assumed to be pixel coordinates (0,0 = the upper left corner of the screen).

Field	Possible Values	Description
X	Positions (Left, Center, Right), Numeric string values (e.g., "10"), Relative values (e.g., "10%"), Attribute references.	Sets the anchor position along the horizontal axis.
Y	Positions (Top, Center, Bottom), Numeric string values (e.g., "10"), Relative values (e.g., "10%"), Attribute references.	Sets the anchor position along the vertical axis.
XAlign	Left, Center, Right	Determines the horizontal positioning of the defined frame in relation to the X and Y anchor position.
YAlign	Top, Center, Bottom	Determines the vertical positioning of the defined frame in relation to the X and Y anchor position.

The XAlign and YAlign fields then allow the positioning of the frame in relation to the X and Y coordinates. For example, the figures below display a circle drawn at the anchor position designated by the X and Y properties (in this case, both X and Y are set to "center"), and a rectangle indicating the Frame area. Setting both the XAlign and YAlign fields to "center" would position the frame so that it is centered on the anchor position (Figure 1). The "left" option in the XAlign field paired with the "top" option in the YAlign field would result in the Frame being positioned on the screen such that the top left corner of the frame was positioned on the anchor coordinates (Figure 2). The "left" and "center" options in the XAlign and YAlign fields respectively would result in the configuration illustrated by Figure 3.

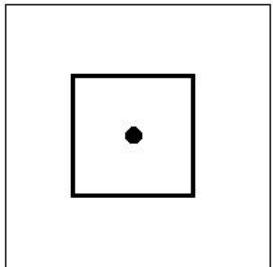


Figure 1. XAlign and YAlign set to "center".

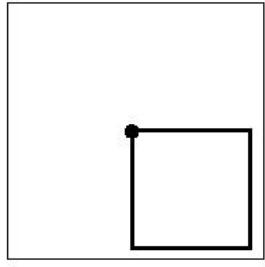


Figure 2. XAlign set to "left", YAlign set to "top".

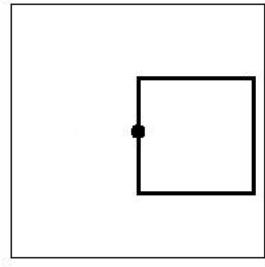
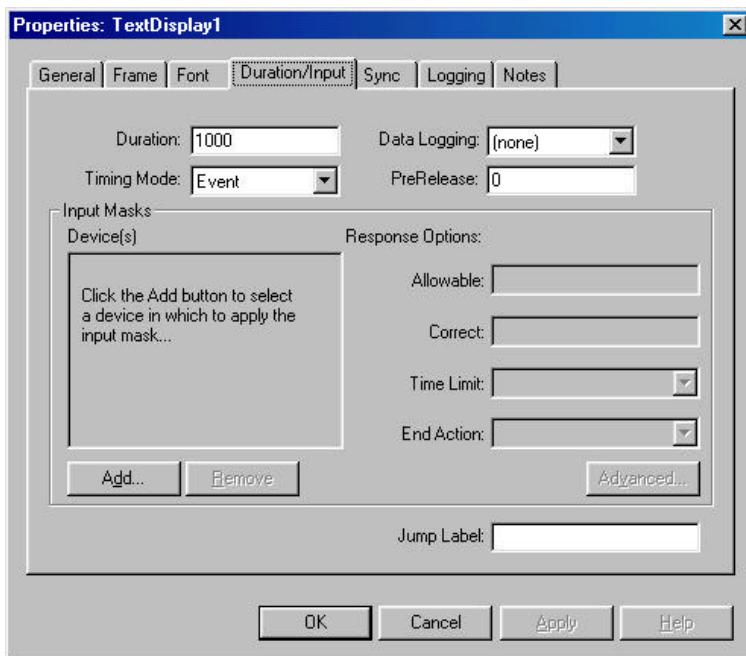


Figure 3. XAlign set to "left", YAlign set to "center".

The Border fields define the appearance of the border drawn around the defined display area. The border is drawn from the outside edge inward, thus a Frame size of 100% can still display a border. The color of the border may be chosen from the available options in the dropdown box in the BorderColor field, or by specifying RGB values (e.g., 255,255,255). The BorderWidth may be set as a positive numeric value, or by using an attribute reference.

Field	Possible Values	Description
BorderColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	Sets the color of the border to be drawn around the display area.
BorderWidth	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Sets the width of the border.

1.3.1.2 Duration/Input Tab





The Duration/Input tab is common to objects capable of collecting input. It is used to set the properties related to the duration of the object, the collection of input, and logging. Input is enabled for an object through the definition of Input Masks. Input Masks define the device to be used for data collection, as well as the specific properties related to that input. Once enabled, a device maintains its own Response Option settings as a mask on the device. Selection of a device in the Device(s) field will display the options for the mask in the Response Options.

Within a single object, multiple masks may be defined to allow input from multiple devices, or to vary the types of allowed responses and resulting actions using a single device for input. For example, masks may be defined to allow input from both the mouse and the keyboard. Or, several different masks may be defined to use the keyboard device. Masks using a common device may differ in the keys designated as allowable input, or in the action to be taken when a response is detected.

Field	Possible Values	Description
Duration	Integer values, Attribute References	Sets the intended duration (msec) of the object's Run method. This setting depends upon the object's End Action setting. The "-1" value may be used to indicate that the object should run indefinitely until the conditions for termination are met (e.g., until a response is collected, audio file playback terminates, etc.).
Timing Mode	Event, Cumulative, Custom	Sets the timing method to be used by the object. <ul style="list-style-type: none">◆ Event - Timing is maintained relative to a single event.◆ Cumulative - Timing is maintained relative to a consecutive set of events.◆ Custom – Timing is maintained by the user.
Data Logging	None, Standard, Response Only, Time Audit Only, Custom	Specifies the category of variables to be logged for the object. Refer to section 1.3.1.4 for a complete description of logging properties. <ul style="list-style-type: none">◆ Standard – Dependent Measures, Time Audit◆ Response Only – Dependent Measures◆ Time Audit Only – Time Audit◆ Custom – User-selected
PreRelease	Integer values	Amount of time released during the processing of the current object to allow for setup of the next object.
Device(s)	Keyboard, Mouse, Serial Response Box, Port Device	Displays devices currently enabled to accept input. Response options (i.e., input masks) must be set for each device separately. A single device may have several input masks.
Allowable	Alphanumeric values, Function Keys, Attribute references, {ANY}, {ALPHA}, {NUMBER}	Defines the allowed input for the specific device. Any character entered into the Allowable field becomes a valid response key. Function and special keys are enabled for input using left and right brackets (e.g., {F1}, {SPACE}). Keys not specified in Allowable are essentially ignored.
Correct	Alphanumeric values, Attribute references	Value used to score the response collected by the input object. If there is no correct response, the Correct field may be left blank.
Time Limit	(same as duration), (infinite), Integer values, Attribute references.	Specifies the amount of time allowed for extended responses. The Time Limit may extend beyond the Run duration of the object.
End Action	(none), Terminate, Jump	Determines the action, if any, to take upon termination of the input mask. Care must be taken when using End Action in conjunction with extended input (i.e., Time Limit). If the Time Limit extends past the Run duration of the object, setting End Action to "terminate" will result in termination of the currently running object, which may or may not be the object collecting the input.
Jump Label	String values	Identifies a Label object within the current Procedure to which program execution jumps when input is received on a mask that has an End Action of "Jump".

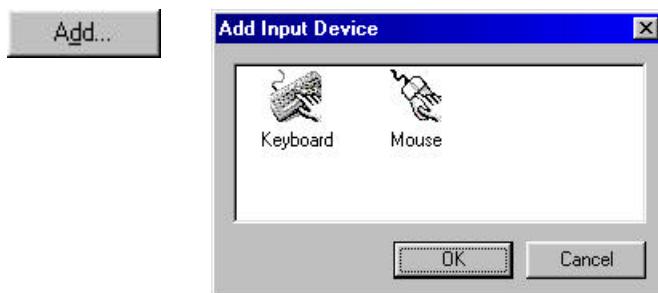
Special mention is warranted by the Jump Label property. The user should take care to understand that the Jump Label is independent of any input masks enabled, and simply serves to



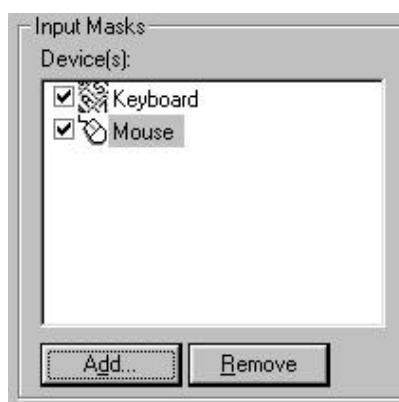
initialize or update the current jump destination. Therefore, while set by a specific object, the Jump Label simply indicates the location to which processing will move if a jump occurs. Jump Label is not specific to the object setting Jump Label, and the jump may occur in relation to another object completely. Once initialized, the Jump Label setting remains active until updated by another object, or until the end of the Procedure. Use of the Jump Label property requires the use of a Label object within the same Procedure.

Adding Devices

When a new object is created, input is NOT automatically enabled for that object. To enable input for an object, an input mask must be defined, enabling a specific device and response options for that device. Devices may be added to the Devices field using the Add button. Clicking the Add... button displays the Add Input Device dialog, listing the objects currently available as input devices. The devices available for selection from this dialog are determined by the devices initialized in the Experiment object. By default, the mouse and keyboard are initialized in the Experiment object as valid input devices. Input may also be enabled via other devices (e.g., PST Serial Response Box, Port Device, etc.). Refer to the Experiment Object Devices tab documentation for information concerning adding and removing devices globally (section 1.3.2). To select a device to add to the Device(s) list, click the specific device icon in the Add Input Device dialog, and click OK.



Separate devices may be chosen as input masks for different objects within the same experiment (e.g., one object may receive input via the keyboard and another object may receive input via the mouse), or input from more than one device may be enabled by the same object (e.g., a single object may accept both keyboard and mouse input). For example, as is illustrated below, the Input Mask devices for an object may be set to collect a response from both the mouse and the keyboard.





Likewise, a single device may be enabled more than once in order to collect input according to different parameters (i.e., different Input Mask settings). The settings for different input masks are independent, thus the Response Options must be set for each device listed in the Device(s) list. To view the property settings for a specific mask, select the device in the Device(s) list, and the properties for that mask will be displayed in the Response Options fields. The independent setting of properties allows a great deal of flexibility and control in the flow of the experiment. For example, a trial could be terminated when any response has been made (e.g., either a keyboard or mouse response), or perhaps only when a particular key is pressed (e.g., enable multiple Keyboard devices, but set the End Action to Terminate for only one).

Removing Devices

To delete a device from the Device(s) list, select the device by clicking on it, and click the Remove button. When a device is removed, it is no longer listed in the Device(s) list.



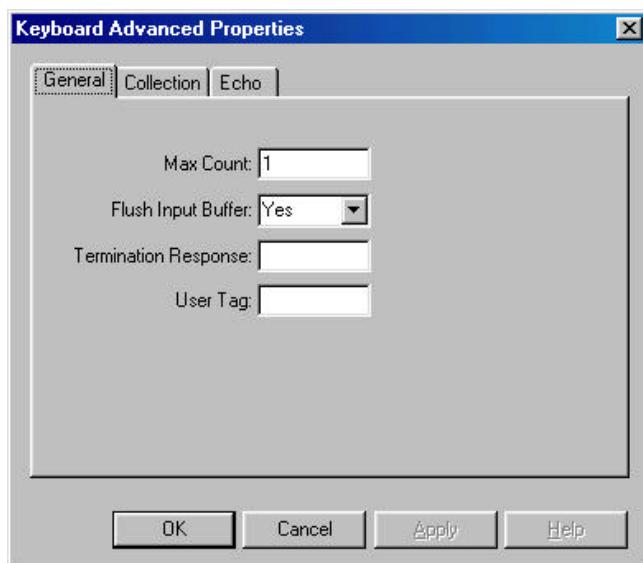
The device may be re-entered using the Add button (see Adding Devices above), but device settings must be entered manually (i.e., settings will be initialized to defaults). In order to disable a device and retain the settings (i.e., not completely remove it from the experiment), click the checkbox to the left of the device name to uncheck the box. When a device is disabled, it remains in the Device(s) list and its settings are retained, but during generation, no script is generated for that device in relation to input.

Advanced Device Properties

The Advanced button allows the user to set additional properties in relation to the collection of input, such as the enabling of multiple responses for string input, the designation of a specific response to terminate an input mask, and flushing of the buffer.

General

The General tab allows the user to enable the collection of string responses (e.g., multiple characters, multiple SRBox button presses, etc.).



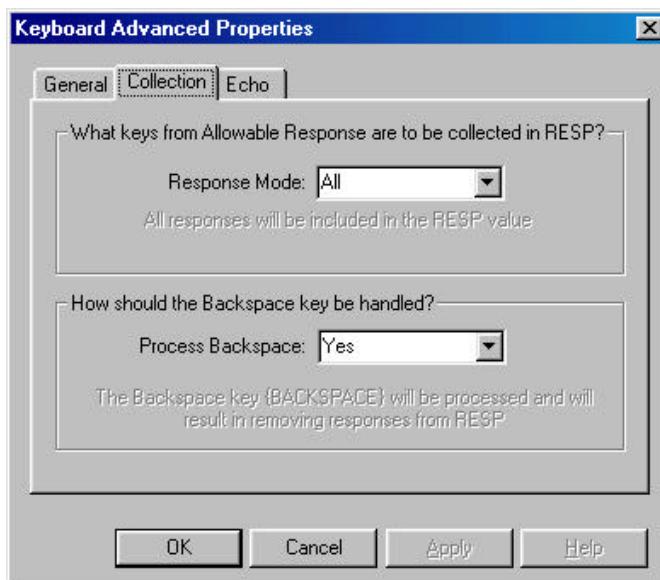


In addition, the properties on this tab allow the user to specify a specific response to terminate the input mask, and whether or not to clear (i.e., flush) responses occurring prior to the arming of the mask from the response buffer.

Field	Possible Values	Description
Max Count	Integer values	Maximum number of responses (e.g., keypresses, SRBox button presses, etc.) expected for input. May be used in conjunction with Termination Response.
Flush Input Buffer	Yes, No	Clears the input buffer to remove any previous responses. If set to "No", negative response times may occur because responses may be accepted that occurred prior to the enabling of the current input.
Termination Response	Alphanumeric	Designates the input to be used to terminate the input mask prior to reaching the maximum response count set in Max Count (e.g., terminate input when {Enter} is pressed).
User Tag	Strings	Defines a string value to be logged with the object. This property might be used to distinguish between responses taken by multiple input masks using the same input device.

Collection

The Collection tab is specific to the keyboard device. This tab allows the user to somewhat restrict the input to be processed during response collection. Regardless of the settings on the Collection tab, all responses are available via script. Refer to the ResponseData object in the E-Basic on-line Help.

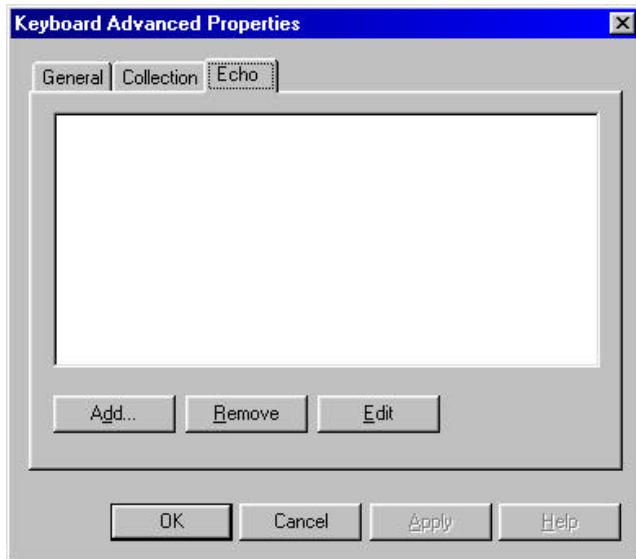


Field	Possible Values	Description
Response Mode	All, Alphanumeric, Attribute references	Limits the types of responses to be collected in the RESP variable. The Alphanumeric option excludes function (e.g., {F1}) and special keys (e.g., {SPACE}).
Process Backspace	Yes, No	Defines the functioning of the Backspace key. If "Yes", Backspace functions normally (delete previous character). If set to "No", Backspace is considered another response. This field is enabled only for Allowable responses including {ANY} or {BACKSPACE}. Inclusion of Backspace in the RESP variable is dependent upon the setting in the Response Mode field.

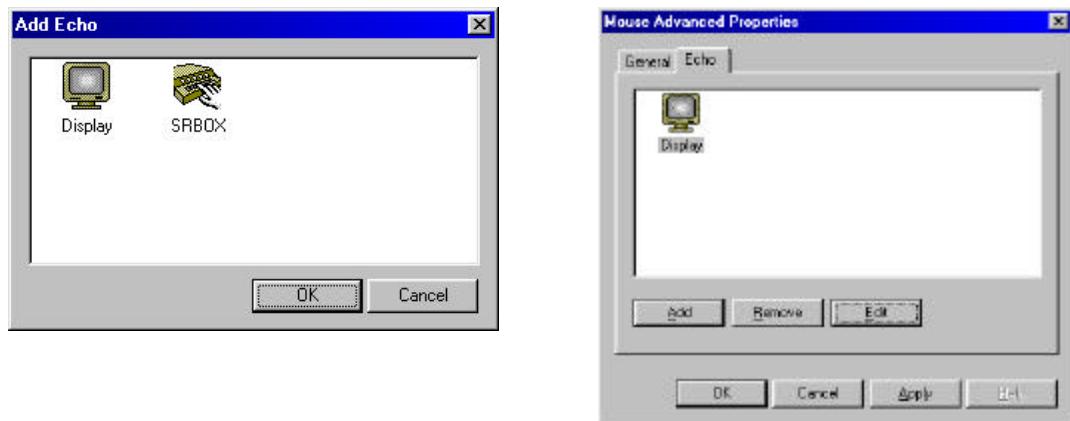


Echo

The Echo tab in the Advanced Properties for input devices allows the specification of properties related to the echoing of input received from the user to an output device (e.g., the visual display or SRBox lamps). The Echo tab permits the user to specify the device to which input will be echoed, and to set properties relevant to that device.



Click the Add... button to display the Add Echo dialog and select the echo device. The devices displayed by the Add Echo dialog depend on the devices enabled in the Experiment object (see section 1.3.2 regarding the Experiment object). The Display device is enabled by default. The dialog below indicates a program in which the user has enabled the SRBox device as well. To select a specific device, click the device icon to select it, and click OK. Regardless of the devices that may be enabled by the Experiment object, only devices offering a method for echoing input (e.g., Display, SRBox) are displayed for selection from the Add Echo dialog.

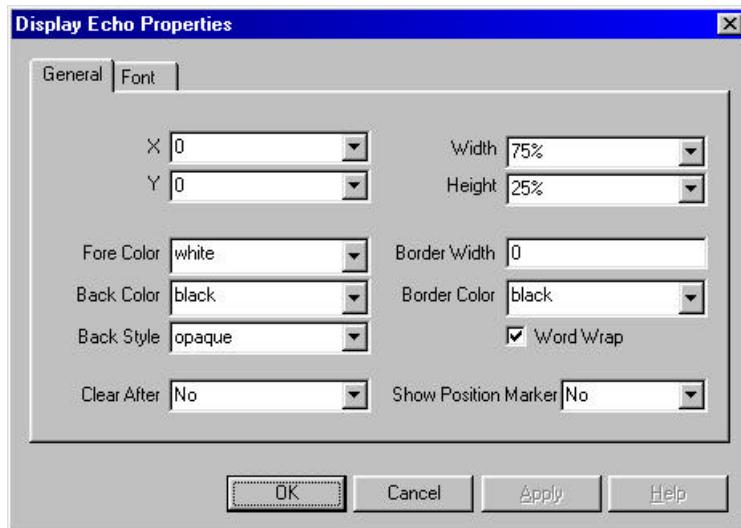


After selection of a device for echo, that device icon is displayed in the Echo dialog. Properties relevant to the echo of input, and specific to the device may be set by selecting the device icon and clicking the Edit button.



Display Echo Properties - General

The properties assignable for formatting the echo of input to the active display relate to the visual characteristics of the input. General properties define the characteristics of the area designated for input.

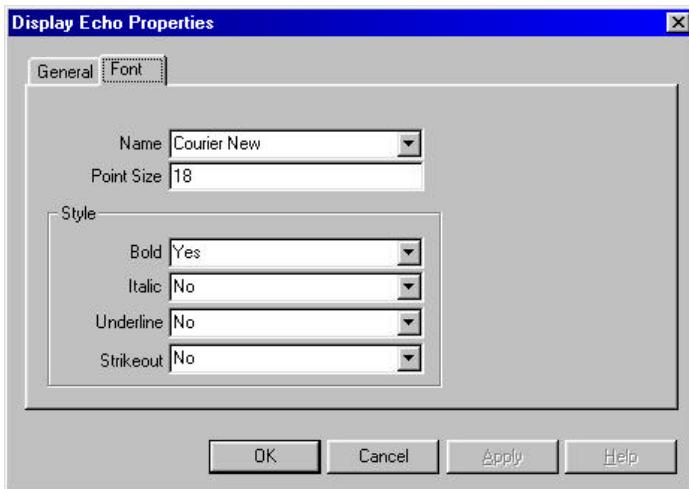


Property	Possible Values	Description
X	Numeric string values (e.g., "10"), Relative values (e.g., "10%"), Attribute references.	Sets the anchor position along the horizontal axis.
Y	Numeric string values (e.g., "10"), Relative values (e.g., "10%"), Attribute references.	Sets the anchor position along the vertical axis.
Width	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Sets the width of the display area to a percentage of the x (horizontal) resolution, or to a fixed pixel size.
Height	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Sets the height of the display area to a percentage of the y (vertical) resolution, or to a fixed pixel size.
ForeColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	Sets the ink color for the display of text in the Text field.
BackColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	The background color on which to display the text characters (functionality depends on BackStyle property).
BackStyle	Opaque, Transparent	Defines the style of the background on which the text is displayed.
Border Color	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	Sets the color of the border to be drawn around the display area.
Border Width	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Sets the width of the border.
Clear After	Yes, No	Controls the clearing of the defined display area after the object's termination condition is met.
Word Wrap	Checked, Unchecked	Determines whether input extending beyond the width of the input area will be wrapped to the next line.
Show Position Marker	Yes, No	Determines whether or not a cursor marks the location at which the next input will be displayed.



Display Echo Properties – Font

The Font tab defines the properties related to the size, color, and style of the text to be echoed to the active display.



Field	Description	
Name	Specifies the name of the font to be used for the display of text.	
Point Size	Specifies the size of the font.	
Style	Bold	Sets the bold status of the text.
	Italic	Sets the italic status of the text.
	Underline	Sets the underline status of the text.
	Strikeout	Sets the strikeout status (i.e., line drawn through the text).

SRBox Echo Properties

The properties assignable for formatting the echo of input via the SRBox relate to the control of the lamps.

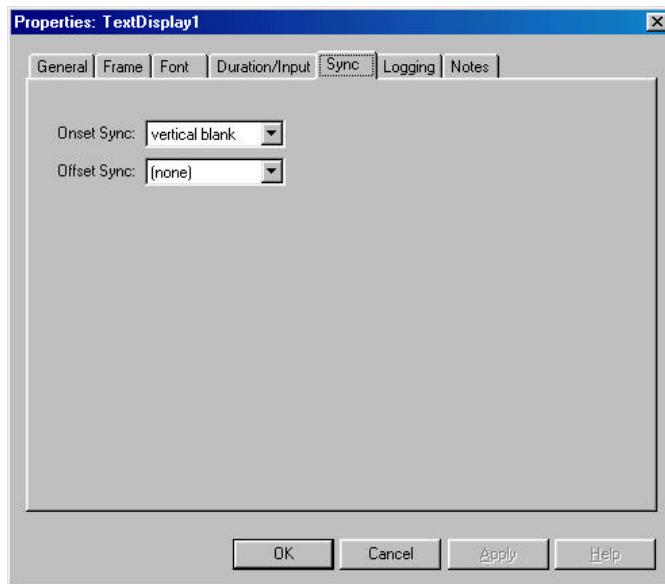


Field	Possible Values	Description
Lamp Mode	Normal,	Turn lamps on for presses, off for releases.
	Toggle	Upon press, toggle state of lamp.
	Sticky	Upon press, turn lamp on and do not turn lamp off. Subsequent presses do not affect the state of the lamp.
Clear After	Yes, No	Controls the clearing of the defined display area after the object's termination condition is met.



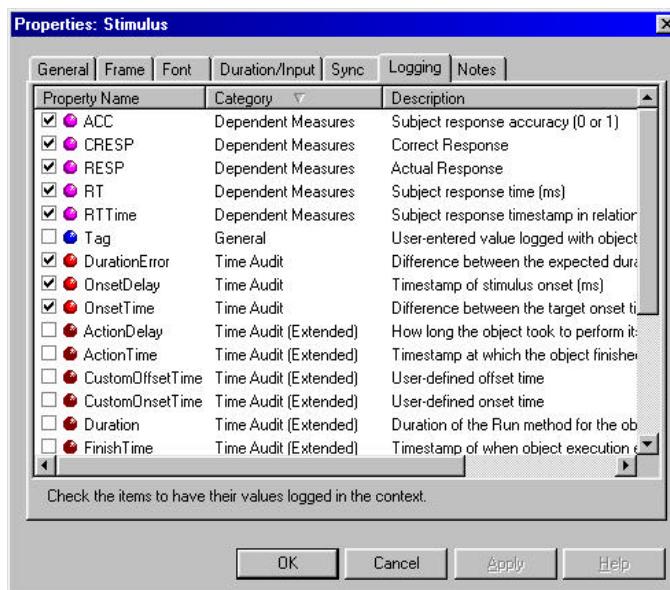
1.3.1.3 Sync Tab

The Sync tab allows the user to control synchronization of an object's onset and offset with specific events (e.g., the refresh of the screen). By default, the onset of an object is set to synchronize with the refresh of the display, and no synchronization is set for the offset of the object. Onset refers to the beginning of the object's critical action execution (e.g., drawing to the screen). Offset refers to the beginning of the object's offset action (i.e., clean-up).



1.3.1.4 Logging Tab

The Logging tab allows the user to select specific attributes to log for each object. On the Logging tab, the automatic measures available within E-Prime are listed and categorized (i.e., Dependent Measures, General, Time Audit, etc.). Click the checkbox next to each attribute to set the logging status.





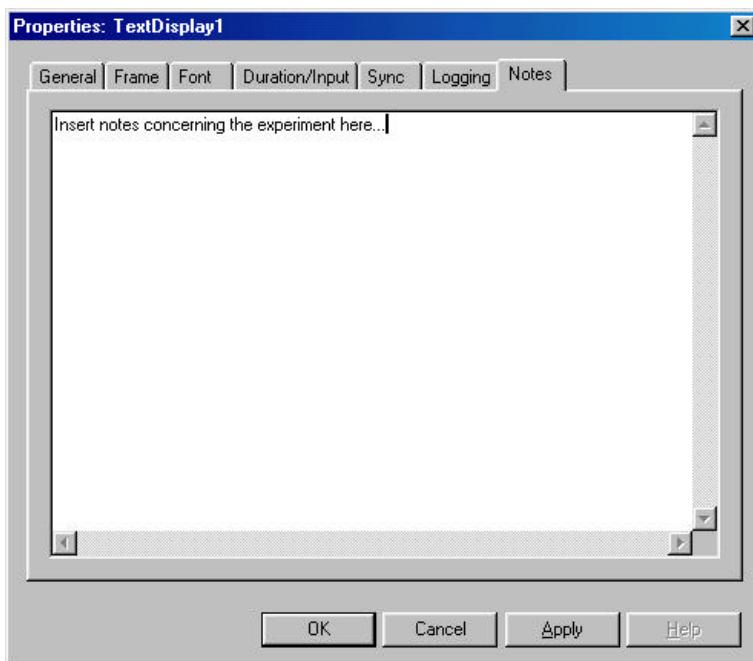
The following table defines the measures that may be automatically logged for an object collecting input. An entire category of measures may be selected through the Data Logging field on the Duration/Input tab.

Property	Category	Description
ACC	Dependent Measures	Reflects the accuracy of the response logged by the input object. ACC is based on a comparison of the RESP and CRESP properties.
CRESP	Dependent Measures	Returns the correct response associated with the input collected by the object. This property is generally set internally according to the value in the Correct field in the Response Options for an object, but may be set via script at run-time.
RESP	Dependent Measures	Returns the last (i.e., for single response input) or entire response (i.e., for multiple response input) collected by the object.
RT	Dependent Measures	Returns the reaction time of the last input collected by the input object, timed relative to the start of the input.
RTTime	Dependent Measures	Returns the reaction time of the input relative to the start time for the experiment.
Tag	General	Defines a string value to be logged with the object. This is a generic property to allow the user to associate some data with the object.
DurationError	Time Audit	Difference between the Duration property setting and the actual duration.
OnsetDelay	Time Audit	Difference between the TargetOnsetTime and the actual OnsetTime.
OnsetTime	Time Audit	Timestamp at which the object begins its critical action (e.g., drawing to the screen).
ActionDelay	Time Audit (Extended)	Amount of time necessary for the object to perform its critical action (e.g., drawing to the screen).
ActionTime	Time Audit (Extended)	Timestamp at which the object finished performing its critical action (e.g., drawing to the screen).
CustomOffsetTime	Time Audit (Extended)	Sets the target offset time when custom timing is in use.
CustomOnsetTime	Time Audit (Extended)	Sets the target onset time when custom timing is in use.
Duration	Time Audit (Extended)	Determines the intended duration of the object in milliseconds.
FinishTime	Time Audit (Extended)	Timestamp at which an object has finished the execution of its Run method.
OffsetDelay	Time Audit (Extended)	The difference between the TargetOffsetTime and the actual OffsetTime.
OffsetTime	Time Audit (Extended)	Timestamp at which the object's offset action (i.e., clean-up) began executing.
PreRelease	Time Audit (Extended)	Time released during the duration of the current object (after the object finishes its critical action) to allow for setup of the next object.
StartTime	Time Audit (Extended)	Timestamp at which execution for the object began (i.e., time at which the Run method began to execute).
TargetOffsetTime	Time Audit (Extended)	Calculated timestamp used by an object as the target time for offset (i.e., begin clean up action).
TargetOnsetTime	Time Audit (Extended)	Calculated timestamp used by an object as the target time to begin performing its critical action (e.g., drawing to the screen).
TimingMode	Time Audit (Extended)	Sets the timing method to be used by the object. Cumulative timing is used to absorb the processing time between events, and to maintain a certain interval duration between events or between trials.



1.3.1.5 Notes Tab

The Notes tab features a text box allowing the user to enter notes. The Notes tab is optional, but it is highly recommended that the notes feature be used to more effectively communicate with a colleague, to record information specific to the Procedure, or to simply serve as a reminder. For example, if passing a program along to a colleague, the Notes tab may be used to describe a specific Procedure in detail, or explain the measures that are being collected.



1.3.2 Experiment Object



The Experiment object as it appears in the Structure view window.

1.3.2.1 Overview

The Experiment object is created automatically for the user when a new experiment specification file is opened. It appears only as the first object in the Structure view, with the experiment name listed in parentheses, and cannot be deleted or moved. Every experiment within E-Prime must have an Experiment object.

1.3.2.2 Interface

The interface for the Experiment object consists of a set of tabbed Property pages enabling the user to set global properties for the experiment, such as the screen resolution to be used when running the experiment, sound properties (i.e., the format for the creation of audio buffers), and available input devices. Furthermore, the Experiment object's Property pages allow the user to enable the Startup Info dialogs (collecting subject number, session number, etc. at run time), and determine the specific information to be collected.

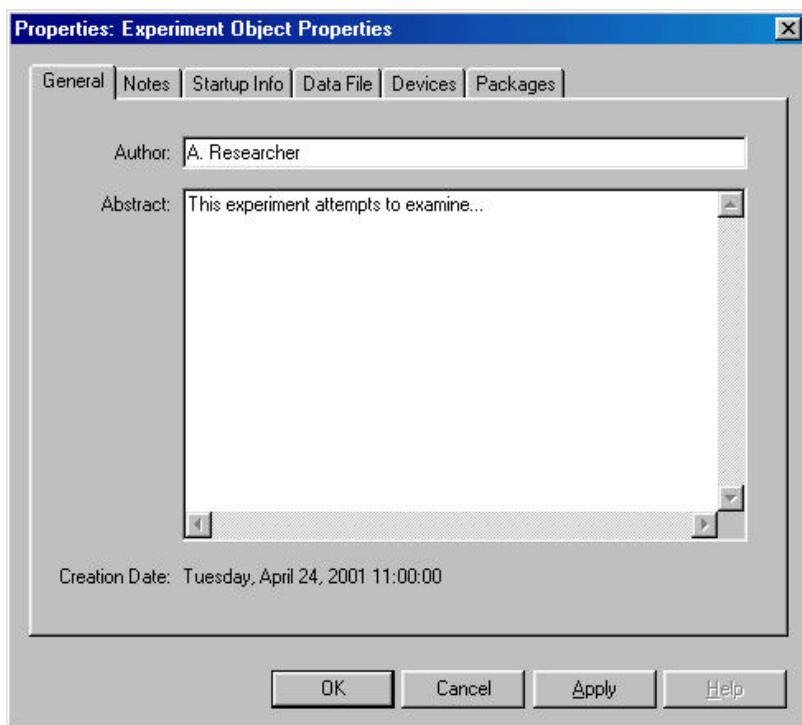


1.3.2.3 Property Pages

The Experiment object Property pages may be displayed by double clicking the Experiment object in the Structure view, or by selecting the Experiment option in the Edit menu.

General Tab

The General tab on the Experiment object's Property pages is used to enter information concerning the author of the experiment, and an abstract of the study. This information may be extremely useful when reviewing a study, or passing an experiment on to a student or colleague.



Startup Info Tab

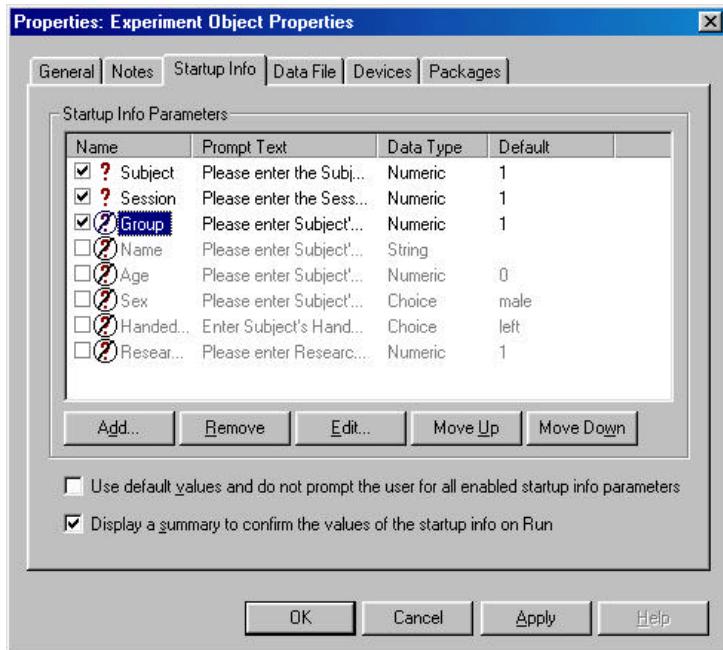
The Startup Info tab is used to define the subject information collected when an experiment is launched. By default, dialog boxes are displayed to collect subject and session number values at the beginning of each run, and a dialog displaying a summary of the information is presented for confirmation.

The startup information may be modified by adding or deleting variables in the Startup Info Parameters field. Alternatively, default values may be used, and dialog prompts may be disabled. While parameters may be added, deleted and/or modified, the Subject and Session parameters are system variables, and must always be present. Therefore, editing of the Subject and Session parameters information is restricted, and the user is not able to either delete or disable these parameters. All parameters enabled via the Startup Info tab are logged as session level variables in the data file (i.e., their values do not vary during the session).



E-Prime Reference Guide

Chapter 1: E-Studio



Field	Description
Name	Name used to log information related to the parameter in the data file.
Prompt Text	Text displayed in the dialog when prompting the user to enter a value for the parameter.
Data Type	Type of data logged by the parameter.
Default (optional)	Value displayed in the dialog when collecting data, or default value used for the parameter if not collected at run time.

In the Name field of the Startup Info Parameters, the checkboxes indicate which parameters will be enabled for the experiment. If the checkbox next to a parameter is checked, the script for the parameter will be generated, and the value for that parameter will be logged in the data file. If the checkbox is unchecked, E-Studio does not generate script for that parameter, and no value is logged. The state of the checkbox determines only whether or not the parameter is enabled, and does not affect the settings for the parameter. Thus, the checkbox permits the user to enable or disable a parameter without having to recreate the parameter, or re-enter the parameter settings.

The question marks next to a parameter name indicate whether a dialog will be presented to collect information at run time, or whether the information will be logged according to the default value.

?	Dialogs will be presented to collect a value at run-time.
?	Default value will be used, and a dialog will not be presented to collect information for the variable.

For example, if a study is being run at various sites, the location of the site might be included in the startup information, and could be set to a default value rather than being entered at run time. In the image above, the checkboxes indicate that Subject, Session, and Group information will be logged for each subject. The question marks indicate that dialogs will be presented to collect only



Subject and Session information, and the Group parameter will be logged according to its default value.

Adding Startup Parameters

Listed Parameters

A list of frequently used parameters is displayed in the Startup Info Parameters field. To add a listed parameter to the data file (i.e., enable the collection of data for the parameter), click the box to the left of the parameter name so that a checkmark appears in the box. Script will not be generated for unchecked parameters, nor will the unchecked parameters be logged in the data file.

If a dialog box is needed to collect a value for the parameter at the beginning of the experiment, click the question mark next to the parameter name in order to remove the circle with the line through it. When the circle with the line through it is present over the question mark, no dialog will be presented to collect information, and the default value will be logged for the parameter. Subsequent clicks on the question mark will toggle between the two states.

To modify the settings (e.g., prompt, default value, etc.) for a parameter, click the parameter name to select it, and click the Edit button, or double-click the parameter name. The Edit Startup Info Parameter dialog will be displayed, allowing the user to edit the settings for the selected parameter. Refer to the *Editing Startup Parameters* section for a complete description of parameter settings.

User-Specified Parameters

Parameters not already listed on the Startup Info page may be added by clicking the Add button below the Startup Info Parameters field.

Add...

The Add... button displays the Edit Startup Info Parameter dialog (see *Editing Startup Parameters*), allowing the user to enter a parameter name, set the parameter's data type and value range, and specify whether to log a default value, or to display a dialog at run time to collect the parameter value. If the parameter value is to be collected using a dialog box at run time, the prompt to be displayed is specified in the Edit Startup Info Parameter dialog as well.

Removing Startup Parameters

To delete a variable from the startup information, select the parameter in the name column and click the Remove button.

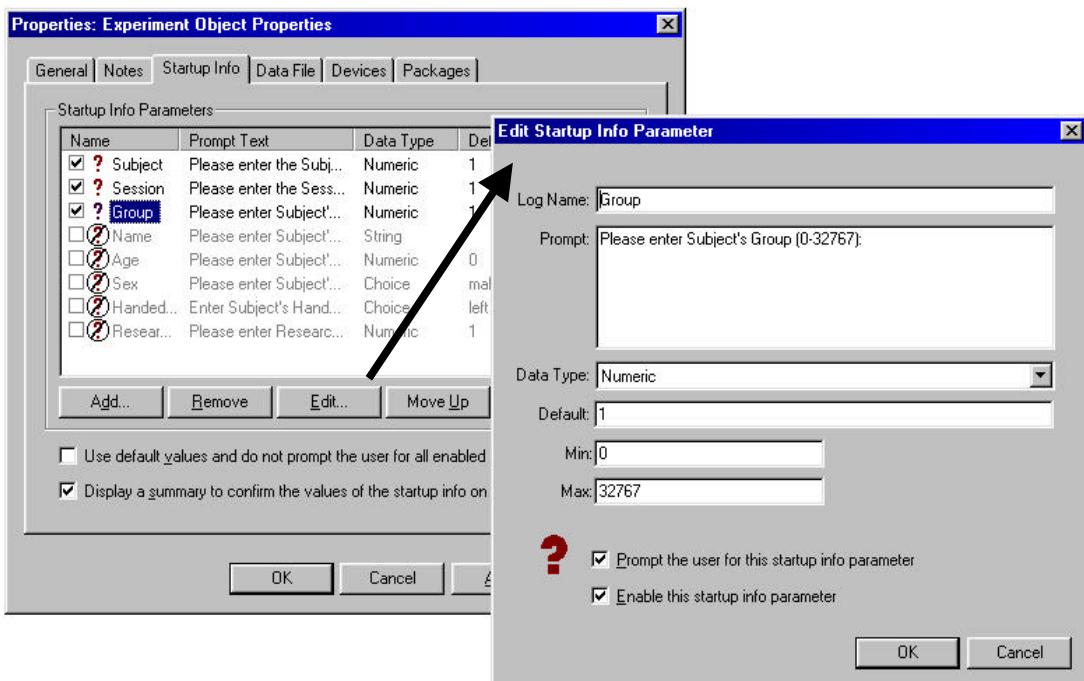
Remove

When a parameter is removed, it is no longer listed in the parameter list. The parameter may be re-entered using the Add... button (see *User-Specified Parameters*), but parameter settings must be entered manually (i.e., settings will be initialized to defaults). In order to disable the parameter and retain the settings (i.e., not completely remove it from the experiment), click the checkbox to the left of the parameter name to uncheck the box. When a parameter is disabled, it remains in the parameter list and its settings are retained, but when the program is generated, no script is generated in relation to that parameter.



Editing Startup Parameters

Settings for existing parameters may be modified using the Edit... button. Select the parameter to be modified in the Name column of the Startup Info Parameters field. When a parameter is selected, the Edit... button is enabled. Click the Edit... button to display the Edit Startup Info Parameter dialog, permitting the user to edit the settings for the selected parameter.

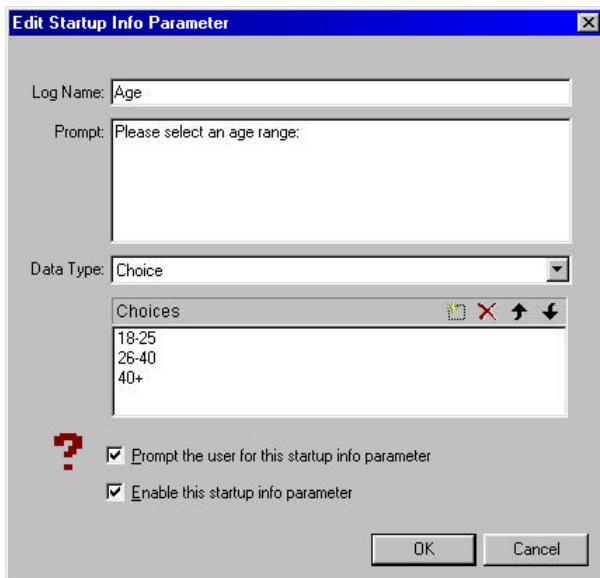


Field	Possible Values	Description
Log Name	String values	Sets the name to be used in the data file when logging values for the selected parameter.
Prompt	String values	Specifies the text to be displayed in the dialog prompting the user to enter a value for the parameter.
Data Type	Numeric, String, Choice	Sets the data type allowed for the parameter.
Default	Integers	Value displayed by the dialog collecting the value at run time, or the value assigned to the parameter when the value is not collected at run time.
Min	Integers	Minimum value allowed for the parameter (Numeric data only).
Max	Integers	Maximum value allowed for the parameter (Numeric data only).
Max Length	Integers	Maximum number of characters allowed (String data only).
Prompt the user for this startup info parameter	Checked, Unchecked	Designates whether to use the default value, or to collect the information via a dialog box presented when the experiment is run. When the box is checked, a dialog will be presented to collect a value for the variable at the beginning of the experiment run. When the box is unchecked, the default value (i.e., the value in the Default field) will be logged for the selected variable. This field may also be set on the main Startup Info page by clicking the question mark next to the parameter name to toggle its state.
Enable this startup info parameter	Checked, Unchecked	Enables or disables the selected parameter. This is equivalent to checking or unchecking the checkbox to the left of the variable name on the main Startup Info page.



Choice Data Type

Startup Info parameters may be defined as accepting one of three data types (i.e., String, Integer, Choice). The Choice data type allows the experimenter to specify a list of choices from which the subject must select. Tool buttons are provided for adding or deleting options, and for rearranging the order in which the options are presented.



Tool Button	Description
	Inserts a line to add an option.
	Deletes the selected option.
	Moves the selected up in the order of the options.
	Moves the selected option down in the order of the options.

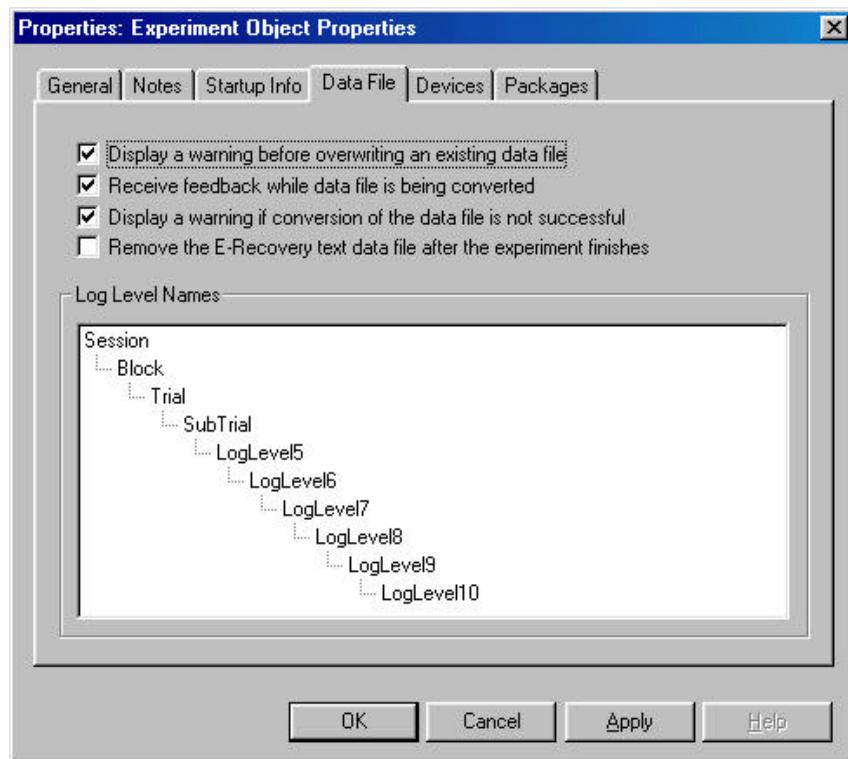
At run-time, the subject is presented with a dialog displaying the choice options. The mouse or the arrow buttons may be used to highlight a selection before accepting it by clicking OK or pressing {Enter}.





Data File Tab

The Data File tab on the Experiment object's Property pages allows the user to set parameters related to the writing of data files, and the names to be used for log levels within the data file. E-Prime's method of writing data involves a two-step process. Data is first written to a text file, then converted to EDAT format after the experiment terminates. Refer to Chapter 6-E-Recovery (this volume) for more information concerning data file conversion.

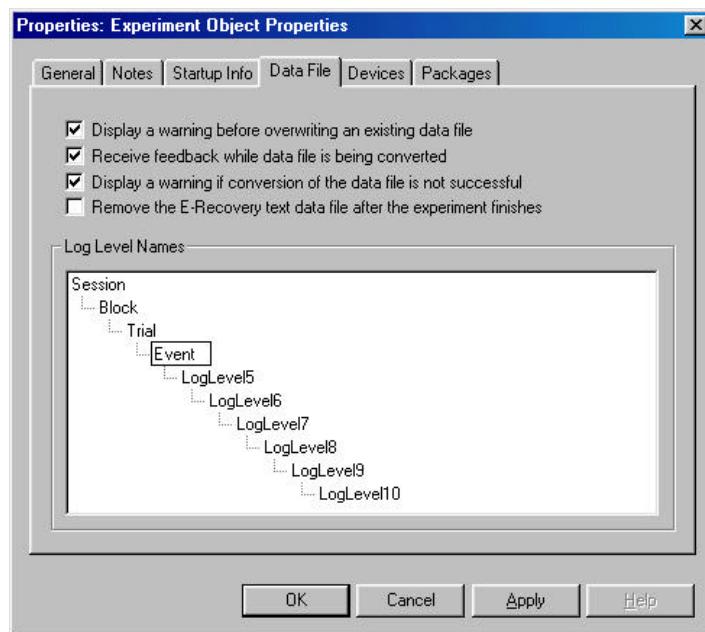


Field	Description
Display a warning before overwriting an existing data file.	Indicates that a file with the same name exists, and prompts the user for confirmation before overwriting the file.
Receive feedback while data file is being converted.	Notifies the user that the data is currently being converted. The dialog will only appear if time permits (e.g., while converting a very large data file).
Display a warning if conversion of the data file is not successful.	Notifies the user that the data file was not successfully converted from the text data file to an EDAT file.
Remove the E-Recovery text data file after the experiment finishes.	Determines whether the text data file is deleted after successful completion to an EDAT file.
Log Level Names	Allows the user to specify names for the logging levels in the data file (see <i>Editing Log Level Names</i>).



Editing Log Level Names

To modify the name by which a level will be logged in the data file, click on the level name in the Log Level Names field and press {F2} (or simply click the level name a second time) to enter edit mode.

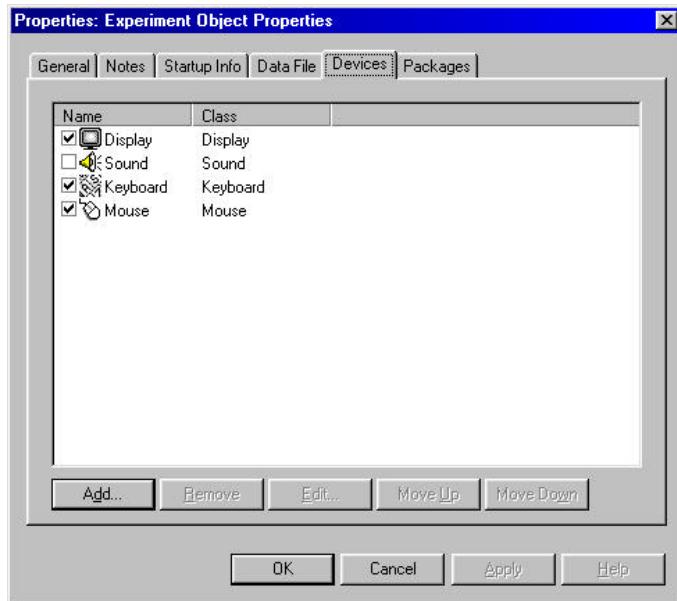


When in edit mode, type the new name for the level and press {Enter}, or click elsewhere in the dialog, to accept the new name. Click OK to accept the new setting and dismiss the dialog. Restrictions for renaming log levels are listed in the table below:

Rules for naming log levels
Cannot rename Session level
Name cannot be empty
Must begin with an alpha character
Permitted characters include a-z, A-Z, 0-9, _(underscore), .(period)
Cannot exceed 80 characters
Cannot conflict with device name (e.g., Keyboard) or device type (e.g., KeyboardDevice)
Cannot be the same as any other log level name
Cannot be the same as any type in the Toolbox (e.g., TextDisplay, Wait, etc.)
Cannot be the name of an E-Object

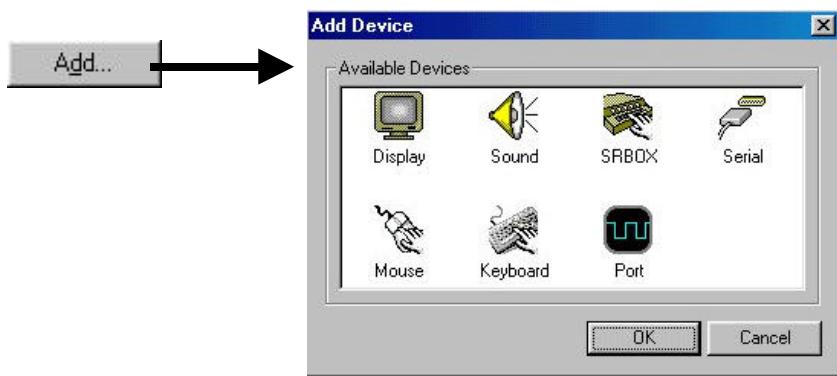
Devices Tab

The Devices tab permits the enabling and initialization of hardware devices to be used in the experiment. By default, the Keyboard, Mouse and Display devices are enabled when a new experiment specification file is created. The Sound device is listed, but not enabled. Devices that are not enabled by the Experiment object will not be available to other objects. For example, to accept input using the PST Serial Response Box, the SRBox must first be enabled via the Experiment object. Once enabled, the device is available to other objects.



Adding Devices

Devices may be added (i.e., enabled) in the Experiment object by clicking the Add... button on the Devices tab. Clicking the Add... button displays the Add Device dialog, from which available devices may be selected.



To add a specific device to the devices list, select the device in the Add Device dialog and click the OK button. When the Add Device dialog is dismissed, the selected device will appear in the devices list, and will be enabled (i.e., the checkbox next to the device will be checked).

Removing Devices

Devices may be removed from the devices list by clicking on the name of the device to select it, and clicking the Remove button.



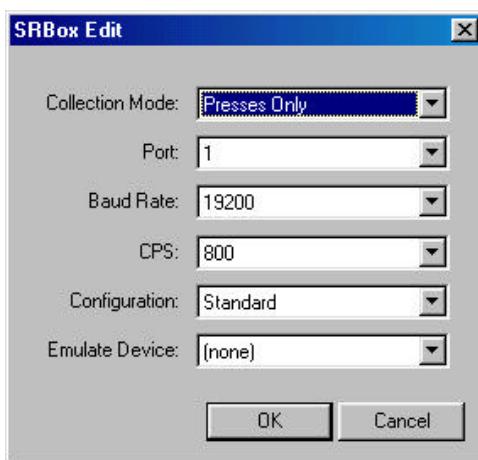
When a device is removed, it no longer appears in the devices list. The device may be re-enabled using the Add... button (see *Adding Devices*), but specific device settings must be entered manually (i.e., settings will be initialized to defaults). In order to disable the device and



retain the settings (i.e., not completely remove it from the experiment), click the checkbox to the left of the device name to uncheck the box. When a device is disabled, it remains in the devices list and its settings are retained, but when the program is generated, the Open command is not called in relation to that device.

Editing Device Settings

The properties or settings for a specific device may be viewed by double-clicking the device in the Name column, or by selecting the device to highlight it, and clicking the Edit button. A dialog box specific to the device allows the setting of properties for that device. For example, it may be necessary to edit the SRBox settings in order to designate the correct Port address.



Specific Device Properties

Device	Property	Possible Values	Description
Display	Width	320, 640, 800, 1024, 1280	Horizontal resolution
	Height	240, 480, 600, 768, 1024	Vertical resolution
	Color Bit Depth	8, 16, 24	Number of colors available to each pixel. 8 bit – 256 16 bit – 65,000 24 bit – 16 million
Sound	Channels	1, 2	Mono or Stereo
	Samples	11025, 22050, 44100	Samples per second (KHz)
	Bits Per Sample	8, 16	Number of bits used to store each sample in digital form.
SRBOX	Collection Mode	Press Only, Releases Only, Presses and Releases	Type of response
	Port	1, 2, 3, 4	Port to which device is installed
	Baud Rate	9600, 19200	Number of times per second a signal transitions between states.
	CPS	800, 1600	Character rate (Characters per second)
	Configuration	Standard, BRU Right Only, BRU Left Only, BRU Unique, BRU Duplicate	Typical E-Prime configuration will only use Standard.
	Emulate Device	(none), Keyboard, Mouse, Port	Permits a device to post its input into the buffer of another device.



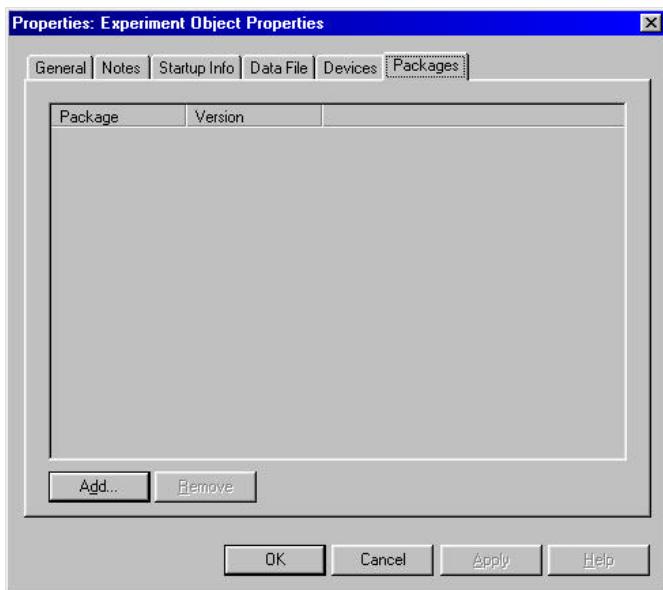
Specific Device Properties Continued...

Device	Property	Possible Values	Description
Serial	Com Port	1, 2, 3, 4	Specifies the identification number of the serial communications port.
	Bits Per Second	110, 300, 600, 1200, 2400, 4800, 9600, 14400, 19200, 28800, 38400, 57600, 115200, 128000, 256000	Specifies the device communication rate.
	Data Bits	5, 6, 7, 8	Specifies the number of bits per byte transmitted and received.
	Parity	None, Even, Odd, Mark, Space	Specifies the parity scheme. Because parity is rarely used, this field can usually be set to None.
	Stop Bits	1, 1.5, 2	Specifies the number of stop bits. 1 is the most common setting.
Mouse	Collection Mode	Presses Only, Releases Only, Presses and Releases	Type of response.
	Open Mode	Direct, Emulate	Determines whether the mouse uses accelerated hardware features of DirectX (Direct mode), or standard Windows messaging (Emulate mode) to retrieve mouse events. In mouse-only setups, Direct mode should be used. Emulate allows most touchscreens to be seen as mouse input.
	Show Cursor	Yes, No	Enables the display of the cursor.
	Emulate Device	(none), Keyboard, Port	Permits a device to post its input into the buffer of another device.
Keyboard	Collection Mode	Presses Only, Releases Only, Presses and Releases	Type of response.
	Caps Lock	On, Off	Determines the state of the Caps Lock at run-time.
	Num Lock	On, Off	Determines the state of the Num Lock at run-time.
	Emulate Device	(none), Mouse, Port	Permits a device to post its input into the buffer of another device.
Port	Collection Mode	Presses Only, Releases Only, Presses and Releases	Type of response.
	Address	0:MAX_LONG	Sets the port address that will be examined for input. The value can be specified in decimal or hexadecimal notation.
	Size	8, 16, 32	Sets the size of the port. This setting will determine how many bits can be read from the port at one time.
	Invert	No, Yes	Sets whether or not the data from the port should be inverted as soon as it is read.
	Mask	MAX_LONG:MAX_LONG	Sets a bit mask to apply to the port after the data is read. The mask is applied by performing a bitwise AND operation between the mask and data. The value can be specified in decimal or hexadecimal notation. It should be noted that a value of "0" will effectively disable inputs from being detected as all bits would always result as 0 when the mask was applied. The value -1 is used as a default since it has no effect on bits.
	Emulate Device		Permits a device to post its input into the buffer of another device.

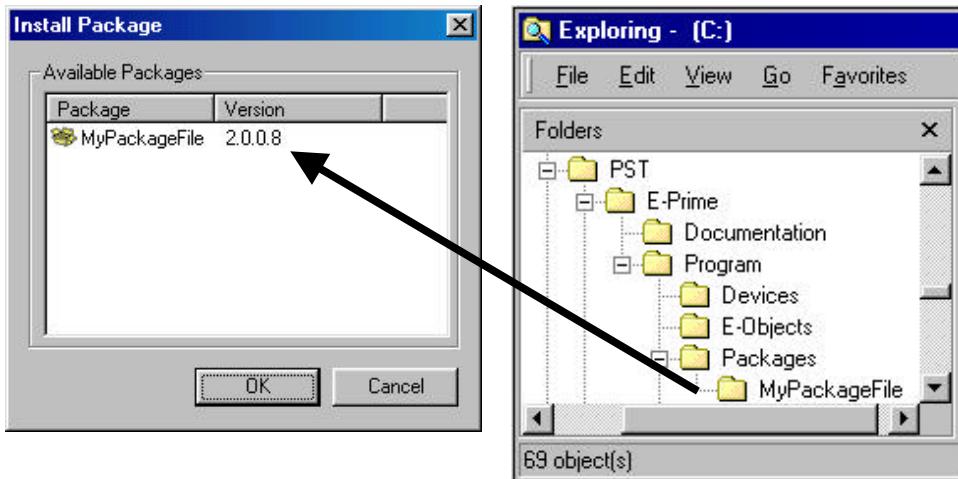


Packages Tab

The Packages tab is used to install or remove an E-Prime Package File, allowing the user to import pre-written script and subroutines into the experiment. Please refer to the PackageCall object documentation (section 1.3.13) for information concerning the use of package files. For a package file to be available, it must first be installed via the Packages tab in the Experiment object properties. By default, no packages are installed, so the Packages tab is blank.

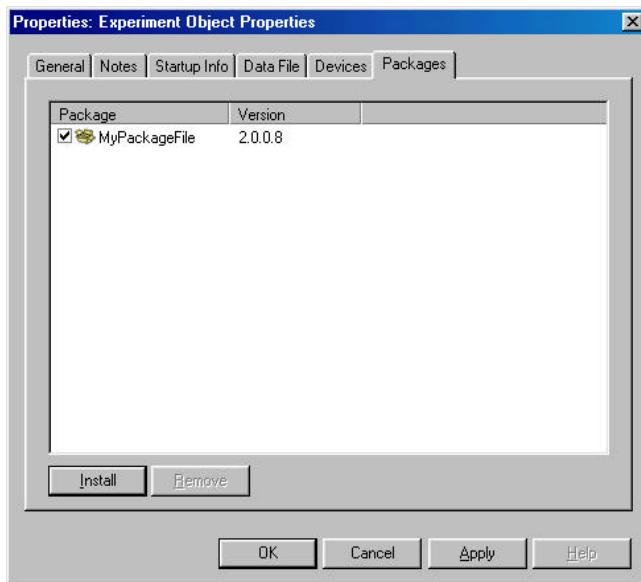


To install a package file, click the Install button, and navigate to the appropriate folder. Each package file must exist in a unique subdirectory within the \Program Files\PST\Prime\Programs\Packages folder. The unique subdirectories are not part of the E-Prime installation, they are user-created, and necessary only when using package files. All files associated with the package file (i.e., EPK, BMP) must be placed within the unique subdirectory. For example, in the example below, the MyPackageFile.EPK and associated images (*.BMP) are located in the \MyPackage folder.





Select the desired package file from the list of available packages, and click OK. Once installed, the package file will appear on the Packages tab, and the checkbox next to the package name will be checked, indicating that the package is enabled. Unchecking the checkbox will disable the package file, but not remove it from the experiment.



1.3.3 Procedure Object



The Procedure object as it appears in the Toolbox.

1.3.3.1 Overview

The Procedure object is used to present a sequence of events. Events occur in sequential order, unless another control mechanism is imposed (e.g., jumps). Experiments are usually comprised of multiple Procedures (e.g., session, block and trial level Procedures) defining the levels of the experiment.

1.3.3.2 Interface

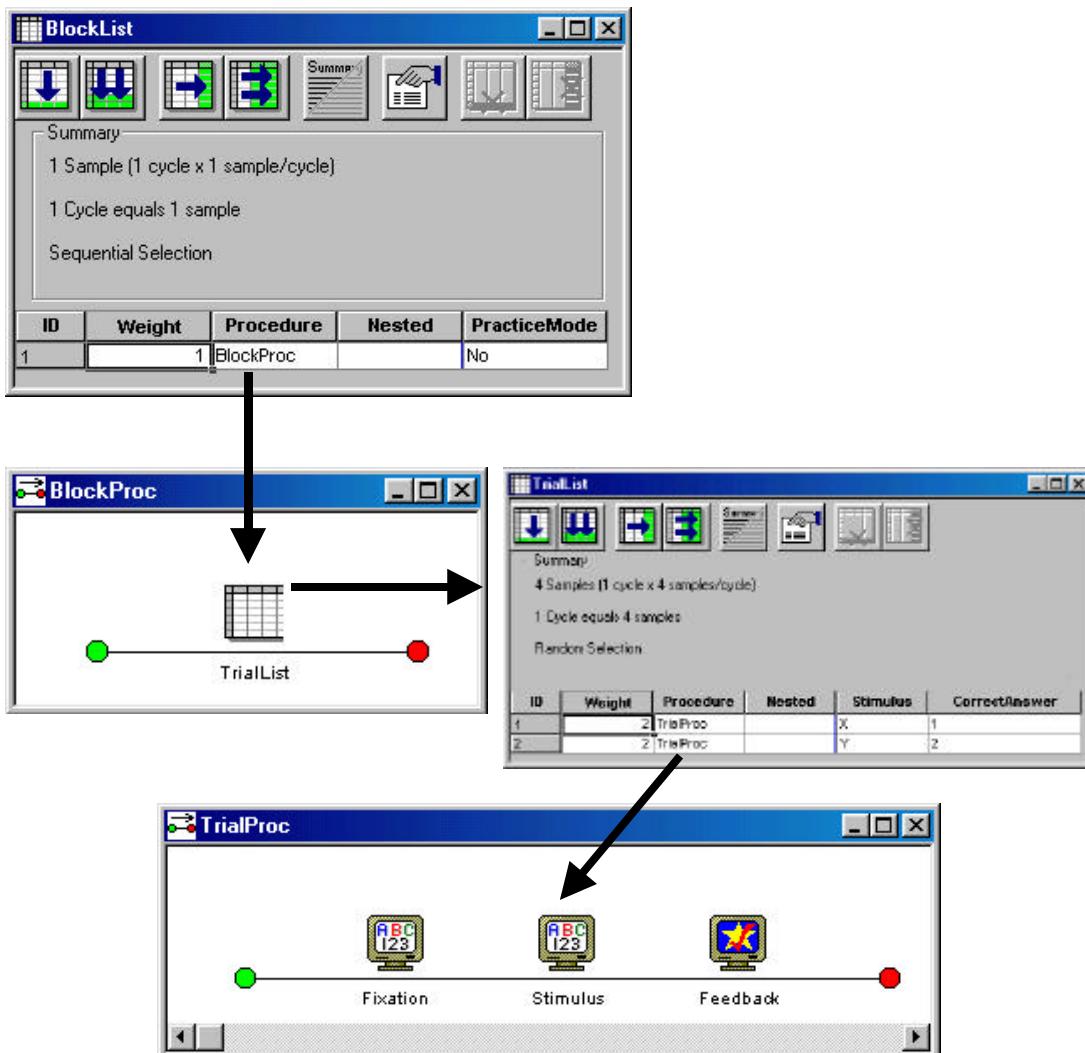
The Procedure object window consists of a single *procedural timeline*. Objects may be dragged and dropped onto the timeline to determine the sequence of events defining the Procedure.





In the image above, the procedural timeline consists of three objects. The order of the objects, left to right, represents the sequence of events for the Procedure. In this example, the first event will be the TextDisplay object named Welcome, followed by the trials run by the TrialList and finally the TextDisplay object called Goodbye.

Procedures can be *nested* (i.e., a Procedure can contain other Procedure objects as part of the timeline). However, Procedures are not referenced directly by another Procedure, and in fact, it is not possible to add a Procedure object directly to a procedural timeline. It is necessary to use a List object to launch a Procedure. For example, as shown in the example below, the BlockList runs the block Procedure (BlockProc), which in turn calls the TrialList to run the trial Procedure (TrialProc).



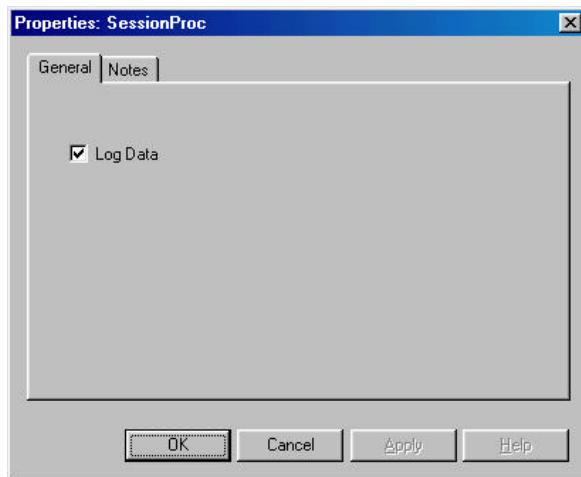
1.3.3.3 Property Pages

Like all objects within E-Studio, the Procedure object features a set of tabbed Property pages. The Property pages are accessible via the Properties window. Click the ellipsis (...) in the Property pages field to display the Property pages. The Procedure object's Property pages contain two tabs: General and Notes.



General Tab

The General tab features a checkbox option to set the status of data logging. When the Log Data box is checked, data will be logged for the Procedure. By default, all Procedures are set to log data.



If the LogData checkbox is unchecked, no data will be logged for the Procedure. For situations in which contingent logging is required, it is possible to log data manually using the Log method of the Context object (i.e., c.Log). See Chapter 1-Using E-Studio in the User's Guide for a description of the Context object.

1.3.3.4 Properties Window

The following are the specific properties of the Procedure object:

Property	Possible Values	Description
(Name)	String values	Represents the name or label of the selected object.
(About)	None	Reveals version information for the Procedure object.
(Property Pages)	None	Reveals the property page interface to specify the properties for the selected object.
LogData	Yes, No	Determines if data is logged for the Procedure.
Notes	String values	User-entered text useful for storing information or comments related to an object.
Tag	String values, Attribute references	User-defined string that is useful for associating information with the object.

1.3.4 List Object



The List object as it appears in the Toolbox.

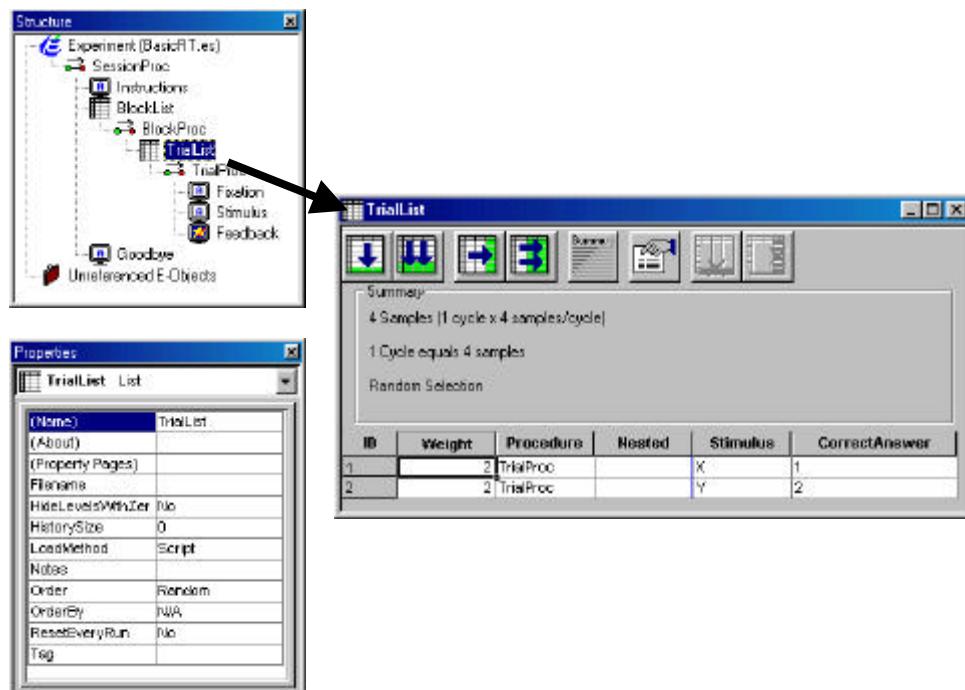
1.3.4.1 Overview

The List object is vital to all experiments in E-Prime. The purpose of the List object is to organize the data to be used within the experiment. The List object is where the independent variables, and their levels are organized. In E-Prime, independent variables are referred to as *attributes*.



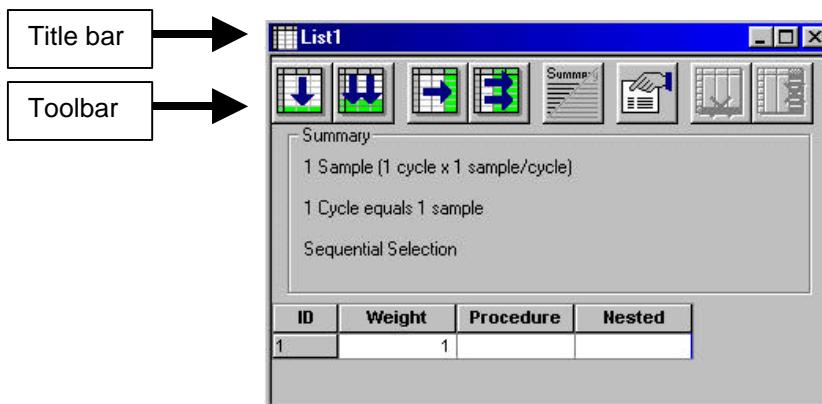
The term 'variable' has a more specific connotation for E-Prime, programmatically speaking (i.e., a variable is something within programming code). Variables are temporary, and are discarded after their use (e.g., a variable may be used as a counter), while attributes are logged in the data file. Please refer to section 4.3.1 in *Chapter 4-Using E-Basic* in the *User's Guide* for a discussion of experimental context, and availability of data within the context.

To create a new List object, click the List object icon in the Toolbox and drag it to the Workspace, to a specific location within the Structure view, or to an existing Procedure. Double clicking an existing List in the Structure view will result in the window representation for that object being opened in the Workspace, and its properties will be displayed in the Properties window.



1.3.4.2 Interface

The List object interface is comprised of a title bar, toolbar, optional summary information and a spreadsheet-like representation of attributes.

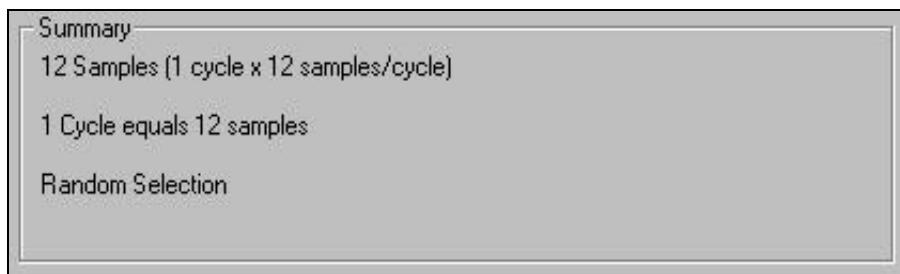




The *title bar* indicates the name of the List object, along with the List icon. The *toolbar* features the available controls for the List object. The controls are also accessible by right clicking in the List object window. The controls available vary based upon the current state of the object. For instance, when an entire row is selected the delete row button becomes active.

Tool Button	Description
	Adds a new row to the bottom of the spreadsheet grid.
	Adds multiple new rows to the spreadsheet at one time.
	Adds a new column to the right of the right-most column in the spreadsheet grid.
	Adds multiple attributes to the spreadsheet at one time.
	Reveals or hides the summary information from the window.
	Opens the set of tabbed Property pages specific to the List object.
	Deletes the selected row or rows from the spreadsheet grid. The Delete Level button is activated only when at least one entire level/row is selected (highlighted). Verification from the user is required before the action is completed.
	Deletes the selected column or columns from the spreadsheet grid. The Delete Attribute button is activated only when at least one entire column is selected (highlighted). Verification from the user is required before the action is completed.

The *Summary* displays the currently specified pertinent properties of the List object. Specifically, the number of samples, the definitions of samples and cycles for the List, and selection information are displayed here. The summary information can be hidden or displayed using the Show/Hide Summary button on the toolbar.



The *spreadsheet grid* features an ID value for each row/level by default. Additionally, the Weight, Procedure, and Nested attributes are also listed by default. These attributes are created by the



system, and may not be deleted. They may, however, be hidden via the View tab (see View tab description below). A blue line separates the system attributes from the user-defined attributes.

ID	Weight	Procedure	Nested
1	1		
2	1		
3	1		
4	1		
5	1		
6	1		



Attribute	Description
ID	Located to the left of each row and may be used to select individual rows, for row-wise editing functions. Rows may also be re-ordered, but they retain their sequential numbering (e.g., if level 5 is moved to the 2nd position, it becomes item 2, and items 2, 3, and 4 become 3, 4, and 5, respectively).
Weight	Indicates the number of times the level is to be counted in the full list (i.e., number of repetitions). For instance, if the Weight of a level is 2, it will be counted twice in the complete list. Rows with Weight 0 are grayed to indicate that they will not be used in the List. Additionally an option can be set to hide these rows within the Property pages of the List object.
Procedure	The Procedure value indicates the name of the object (typically a Procedure object) to be called/executed when that particular row is selected.
Nested	<p>The Nested column allows one list to reference other lists and their attributes. Nested lists are useful when different conditions include different sets of attributes and stimuli, or for sampling from more than one list on any single trial.</p> <p>The values in this cell indicate from which List(s) the attribute values are to be selected. For instance, to select from two lists independently, specify the names of both lists in the Nested column (separated by commas). An example of nested lists is provided in the sample experiments as part of the E-Prime installation.</p>

The List object also features the following controls and features:

Cut, Copy, Paste	Allows easier editing within the spreadsheet. In general, these operations mimic those in Microsoft® Excel (e.g., operates on cells, ranges, rows, columns and uses tab delimited text during clipboard operations).
Disjointed Row Selection	Ability to select rows to allow operations to be enacted upon non-contiguous rows (e.g., set the Procedure to a common value for rows 2, 4 and 6).
AutoComplete	Automatically complete the entry if the first few characters typed match another entry in the column.
AutoFill	Ability to fill columns of a List quickly as in Microsoft® Excel, by selecting cells and dragging fill handles.

1.3.4.3 Property Pages

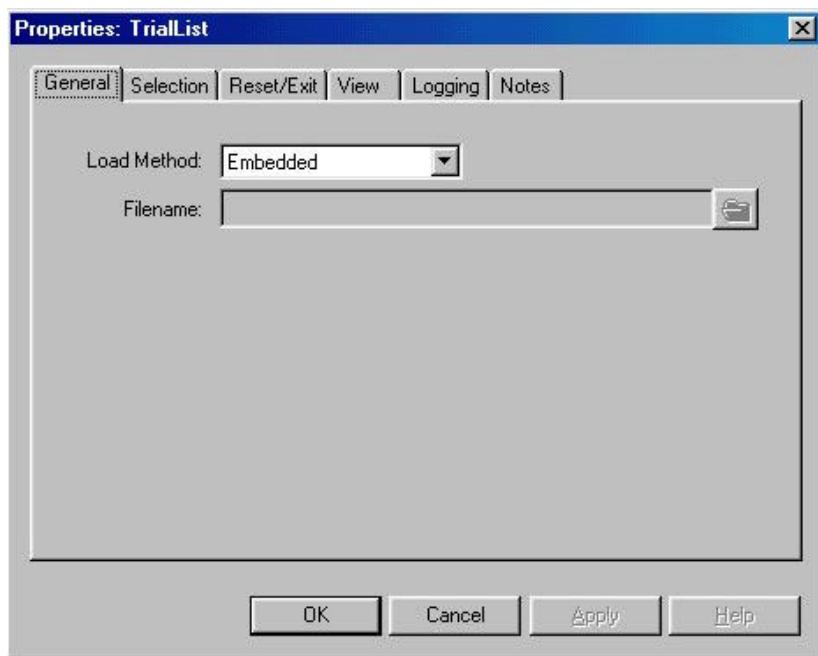
In addition to the Properties window, a set of Property pages is provided for the purpose of setting the List object properties. To access the Property pages for the List object, press the Property Pages button in the List object toolbar.





General Tab

The General tab permits the setting of the Load Method property, to determine how the List object will be populated. There are three Load Method options: Embedded, Script and File. By default, the LoadMethod property is set to Embedded.

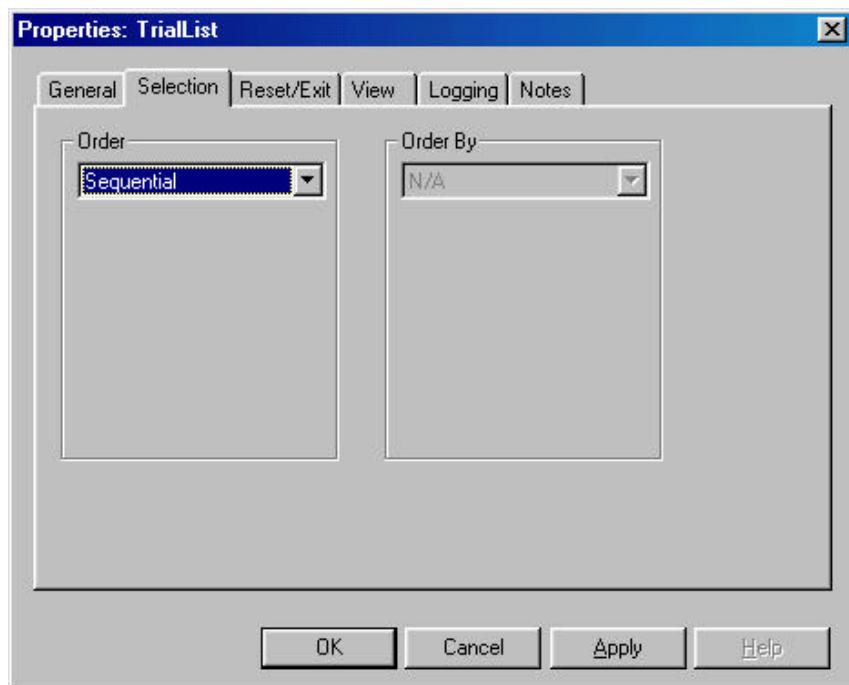


Field	Description	
Load Method	Embedded	Instructs the List to use a "DataSection" to store its values in the full script. With Embedded format, the last lines of the full script contain List defined format of the data. This reduces the size of the script and takes the InitObjects routine much less time to complete.
	Script	The levels of the List will be loaded into the E-Basic script (EBS) file according to the values entered into the List object. This tells the list to use List.AddLevel, List.AddAttrib, and List.SetAttrib calls to fill the List at run-time. The more attributes and levels a List has, the longer the InitObjects routine will take to run.
	File	Instructs the List to load its contents from the file specified by the List.Filename property. The file must be in tab-delimited, ASCII file format, and should contain the level data (i.e., attribute header data). If attributes exist in the file that are not defined in the GUI, these attributes are created. When this mode is selected, the levels in the List object are ignored (and are grayed out). This permits the user to fill in dummy values so that the structure of the experiment remains intact in the Structure view, and/or allows the user to bounce between Load Method values without having to reenter all of the data.
Filename	Existing filenames (txt).	Identifies the name of the text file from which the List will be loaded (see File Load Method above).



Selection Tab

The Selection tab determines the order in which the levels (i.e., rows) within the List are sampled. A variety of ordering options are available through the Order field. Among the options are: Sequential, Random, Random with replacement, Counterbalance, Offset and Permutation.

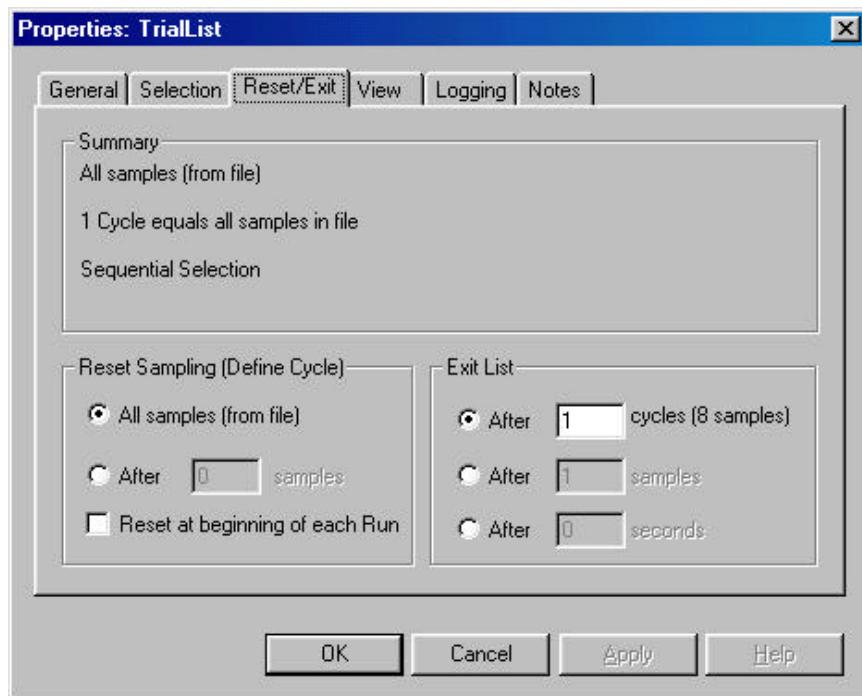


Order	Description	Order By
Sequential	Selection occurs in the order in which the levels appear in the List.	Not Applicable
Random	Selection is based on an internal random number generator. The levels are shuffled at run-time based upon a random number. Random selection occurs <i>without replacement</i> .	Not Applicable
Random With Replacement	Shuffles the levels in the current List object, but replaces each level as it is used. This option will result in the possibility of presenting the same level(s) more than once and other levels being entirely unused.	Not Applicable
Counterbalance	Presents a single level based on the specified option in the Order By field (e.g., Subject = 1 presents level 1). Counterbalancing can be based upon the subject number, session number or group number. If the Order By factor exceeds the number of levels, the modulus (the remainder of the subject number divided by the number of levels returned as a whole number) is used to determine the level.	Subject, Session, Group
Offset	Creates a sequential order that begins at a predetermined offset within the List, wrapping around when it reaches the end. The predetermined offset is based on the Order By factor (e.g., by subject, by session, by group). If the Order By factor exceeds the number of levels, the modulus (the remainder of the subject number divided by the number of levels returned as a whole number) is used to calculate the offset.	Subject, Session, Group
Permutation	Permutation calculates all possible combinations of the conditions using a mathematical algorithm. Then, ordering occurs like Offset, where one of the possible combinations is chosen based on a predetermined offset, and all conditions within that combination are run in sequential order. The predetermined offset is based on the Order By factor (e.g., by subject, session or group).	Subject, Session, Group



Reset/Exit Tab

The Reset/Exit tab provides a method to set the run-time sample and cycle counts for the selected List.

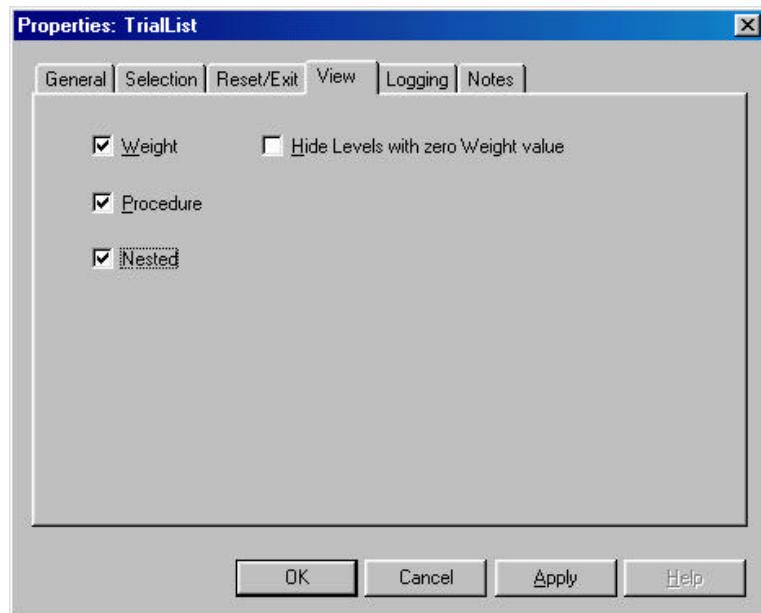


Field	Description	
Summary	Summarizes the current property settings pertaining to selection and definitions of the cycle.	
Reset Sampling (Define Cycle)	After all samples	Resets the list of samples (i.e., replaces exemplars) for re-selection after each has been selected once.
	After (user defined) samples	Resets the list of samples (i.e., replaces exemplars) for re-selection after the specified number of samples.
	Reset at beginning of each Run	Replaces exemplars for re-selection each time List.Run is called. This is helpful if the list is being used more than a single time within an experiment (e.g., practice and experimental blocks).
Exit List	After (user defined) cycles	Terminates the List after the specified number of cycles is completed.
	After (user defined) samples	Terminates the List after the specified number of samples is completed.
	After (user defined) seconds	Examines the expired time before each sample. If the specified time has elapsed, the List terminates.



View Tab

The View tab features a checklist of system items to be displayed in the spreadsheet grid of the List object window. The Weight, Procedure and Nested attributes are created by the system, and may not be deleted. However, it is possible to remove one or all of them from the display of the List by unchecking them on the View tab.



Additionally, it is possible to hide levels (rows) with a weight value of "0". If this option is checked, levels with a weight of "0" will not be displayed in the List. If this option is unchecked, levels with a weight of "0" will appear grayed out, but visible in the spreadsheet.

Logging Tab

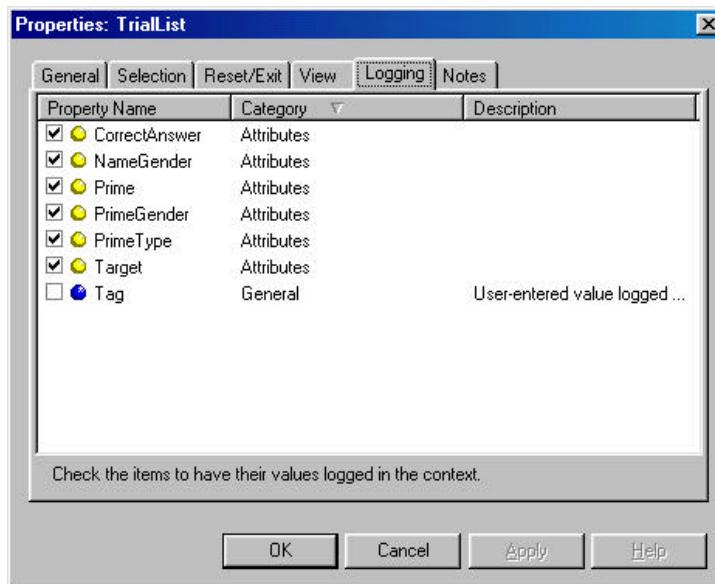
The Logging tab allows the user to select specific attributes to log for each object. The Logging tab for the List object is somewhat different from the Logging tab for an object collecting input. The Logging tab for an object collecting input displays the possible dependent measures collected (e.g., RT, ACC) as well as time auditing measures for the object (e.g., OnsetDelay, DurationError). The primary purpose of the List is as the organization of the data used within the experiment. Therefore, the attributes available for logging on the Logging tab refer to the user-defined attributes entered on the List.

By default, all attributes are logged in the data file. However, it may be useful to turn off the logging for specific attributes if they are not of interest. Click the checkbox next to each attribute to set the logging status. In the image below, CorrectAnswer and Stimulus are attributes defined by the user in the List object. The Tag attribute is defined by the system, but because its value is set by the user, logging may be turned off for this attribute. According to the status of the checkboxes in the image below, CorrectAnswer and Stimulus will be logged in the data file, while Tag will not.

While the disabling of logging for some attributes allows the user to limit the number of attributes logged in the data file, it is highly recommended that all attributes be logged. Many researchers have forgotten or failed to log a measure that was later required by some analysis, or was crucial to some analysis not originally planned. The E-DataAid data handling application includes



filtering and exporting capabilities to easily allow the user to display only those attributes that are desired in the data file. Thus, it is recommended that all attributes be logged, and that E-DataAid's filtering capabilities be used to limit the display or export of the data to include only specific attributes.



1.3.4.4 Properties Window

The List object's properties may also be specified through the Properties window. The following are the properties associated with the List object:

Property	Possible Values	Description
(Name)	String values	Represents the name or label of the selected object.
(About)	None	Reveals version information for the List object.
(Property pages)	None	Reveals the property page interface to specify the properties of the selected object.
Filename	String	Designates the tab-delimited text file from which the List will be populated.
HideLevelsWithZeroWeight	Yes, No	Determines if levels with zero weights are displayed. "Yes" will hide the levels (rows) with the weight equal to zero. "No" still allows the levels (rows) with a zero weight to be displayed, but they will appear grayed out.
LoadMethod	Embedded, File, Script	Determines the method by which the script relevant to the List will be generated.
Notes	String values	User-entered text useful for storing information or comments related to an object.
Order	Sequential, Random, Random with Replacement, Counterbalance, Offset, Permutation	Determines the selection order of the levels.
OrderBy	N/A , Subject, Session, Group	Determines counterbalancing details.
ResetEveryRun	Yes, No	Determines whether to reset the List after each run.
Tag	Alphanumeric	User-defined string useful for associating info with the object.



1.3.5 TextDisplay Object



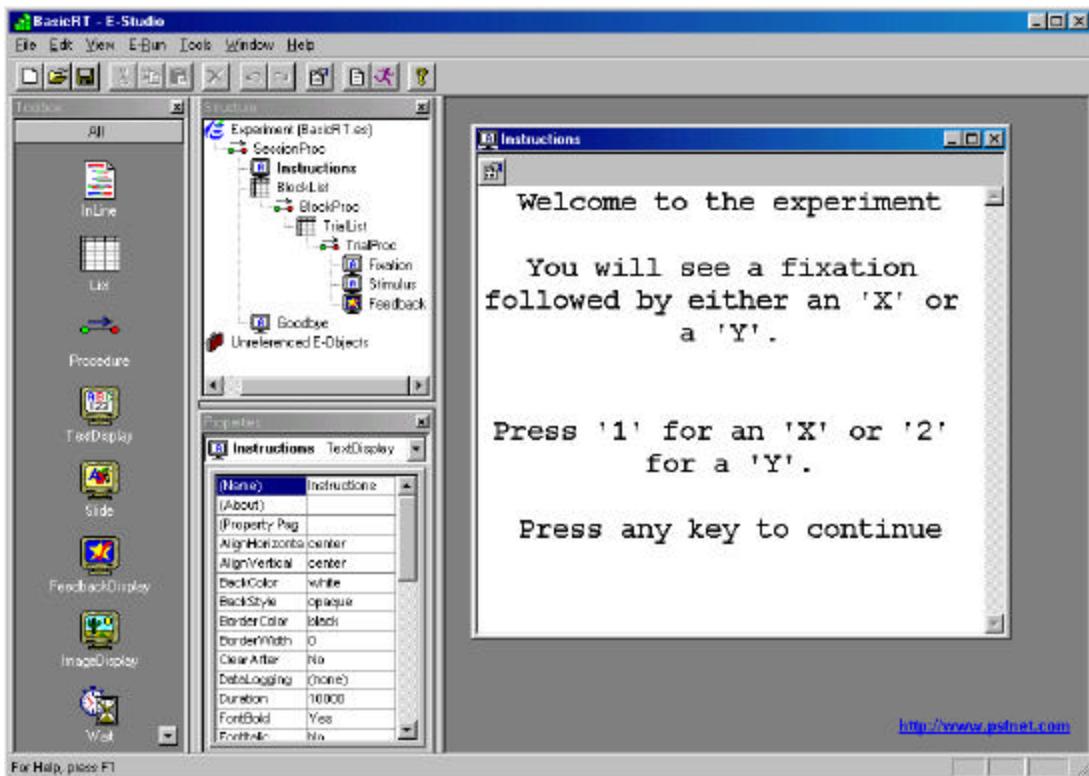
The TextDisplay object as it appears in the Toolbox.

1.3.5.1 Overview

The TextDisplay object is used to organize the display of text to the subject. The TextDisplay is meant to simulate the display that the subject will see, and therefore immediately applies the properties set by the user. For example, in the Text field, the user will see the text displayed in the color specified as the ForeColor property. Likewise, setting the alignment properties to "center" will result in the centering of the text in the Text field. Text is considered a single entity to which properties are applied (i.e., all text is in the same font, forecolor, etc.).

1.3.5.2 Interface

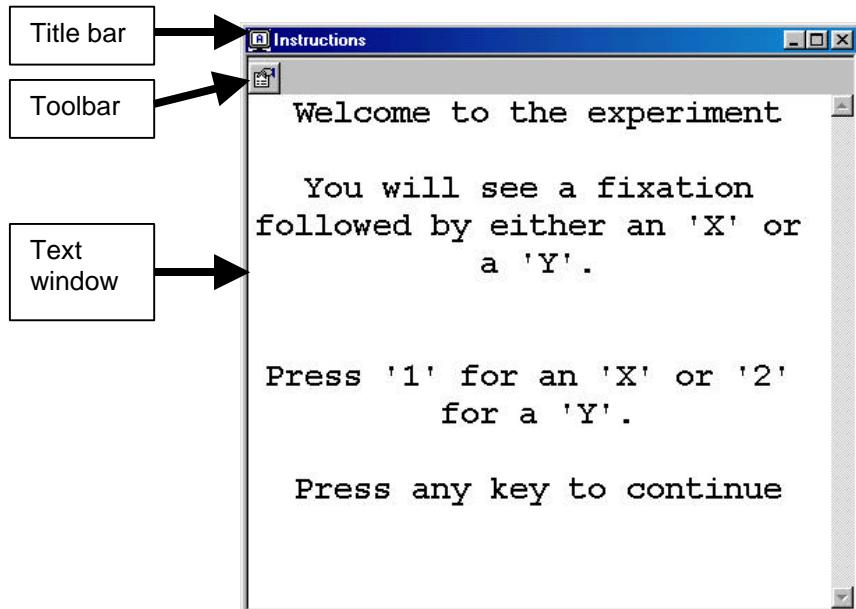
A new TextDisplay may be opened by clicking the TextDisplay object icon in the Toolbox, and dragging the object to the Workspace. For existing TextDisplays, clicking the object once with the mouse in the Structure view to select it will result in the display of that object's properties in the Properties window. Double clicking an existing TextDisplay in the Structure view will result in that object being opened in the Workspace, as well as the properties for that object being displayed in the Properties window.



A TextDisplay object opened in the Workspace displays the name of the object, along with the TextDisplay icon in the title bar at the top of the TextDisplay window. Below the title bar, the toolbar lists the tool buttons associated with the TextDisplay. Currently, the only active tool

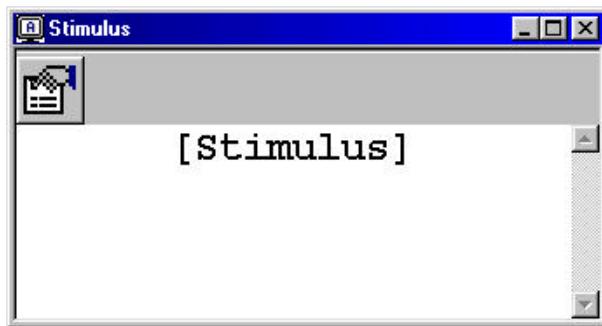


button is the Properties button. The Text window appears at the bottom of the TextDisplay. The Text window is used to enter the text to be displayed at run-time.



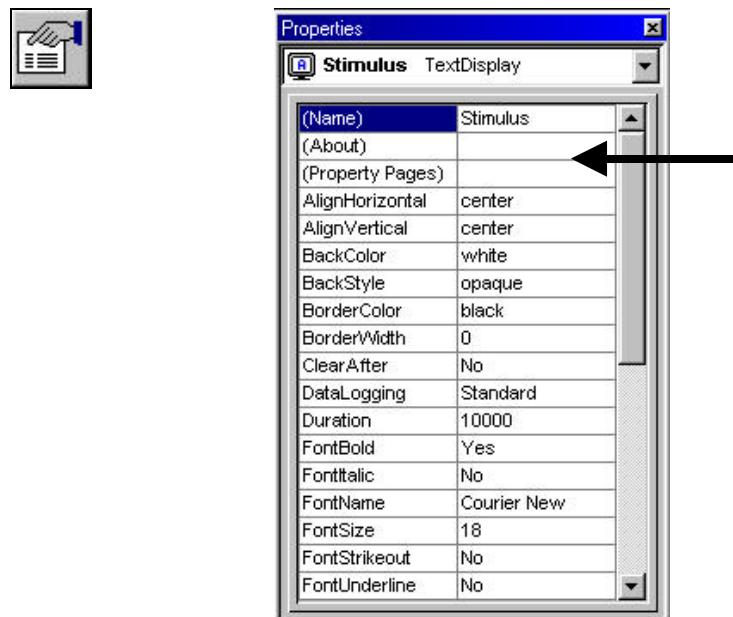
The TextDisplay is meant to simulate the display the subject will see during a run of the experiment. Using the Properties window or the Property pages, the properties of the display may be set. The setting of observable properties (i.e., ForeColor, BackColor, Alignment, Font, etc.) will take effect immediately in the Text window of the TextDisplay.

Constant text may be typed directly into the text window, or a bracket/attribute notation may be used to vary the text to be displayed. The bracket/attribute notation (i.e., [AttributeName]) indicates to E-Run that the information requested is variable, and E-Run will look to the currently running List object to resolve the value of the attribute name in brackets at run time.



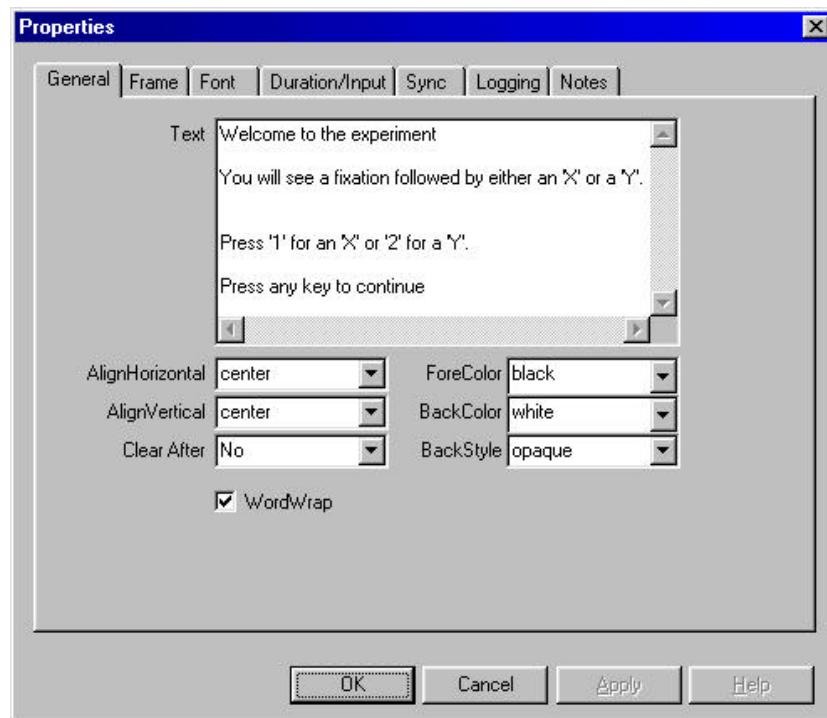
1.3.5.3 Property Pages

The properties for the TextDisplay may be set using the Property pages. The Property pages may be displayed using the Property Pages tool button on the TextDisplay toolbar or by clicking the ellipses (...) in the Property Pages field in the Properties window (select the TextDisplay in the Structure view to display its properties in the Properties windows).



General Tab

The Property pages open by default to display the General tab. The General tab allows the setting of the text to be displayed, and the observable properties of that text. If text has previously been entered into the Text window on the TextDisplay object, this text is reflected in the Text field on the General tab.

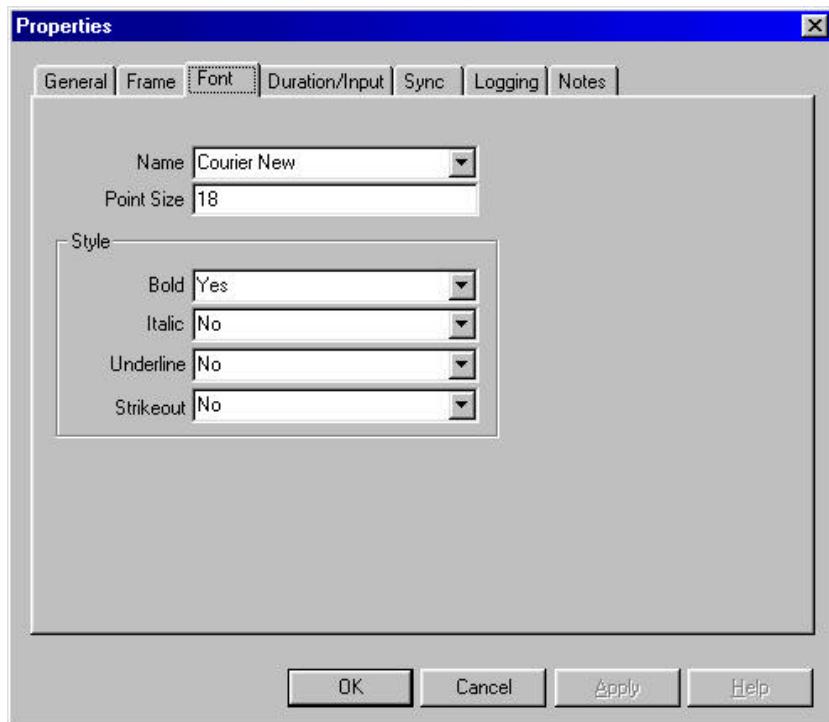




Field	Possible Values	Description
Text	String values, Attribute references	Enter the text to be displayed at run-time.
AlignHorizontal	Left, Center, Right, Attribute references	Defines the horizontal alignment of the contents within the frame (display area).
AlignVertical	Top, Center, Bottom, Attribute references	Defines the vertical alignment of the contents within the defined frame (display area).
ClearAfter	Yes, No	Controls the clearing of the defined display area after the object's termination condition is met.
WordWrap	Yes, No	Toggles the WordWrap feature on or off.
ForeColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	Sets the ink color for the display of text in the Text field.
BackColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	The background color on which to display the text characters (functionality depends on BackStyle property).
BackStyle	Opaque, Transparent	Defines the style of the background on which the text is displayed.

Font Tab

The Font tab allows the setting of the properties related to the font to be used for the TextDisplay. The font name may be typed into the FontName field directly, or may be chosen from the dropdown list of fonts available within the system. The Point Size represents the size of the font and may be typed directly into the field. To access custom fonts, the custom font must be installed on the machine via the Windows Control Panel. The font must be installed on all machines running the experiment using the custom font.





Field	Description	
Name	Specifies the name of the font to be used for the display of text.	
Point Size	Specifies the size of the font.	
Style	Bold	Sets the bold status of the text.
	Italic	Sets the italic status of the text.
	Underline	Sets the underline status of the text.
	Strikeout	Sets the strikeout status (i.e., line drawn through the text).

1.3.5.4 Properties Window

The properties for the TextDisplay may be set using the fields in the Properties window. The following properties are associated with the TextDisplay object:

Property	Possible Values	Description
(Name)	String values	Assigns a user-entered string as the name of the TextDisplay.
(About)	-----	Displays the About TextDisplay dialog.
(Property Pages)	-----	Opens the Property pages for the TextDisplay.
AlignHorizontal	Left, Center, Right, Attribute references	Defines the horizontal alignment of the contents within the frame (display area).
AlignVertical	Top, Center, Bottom, Attribute references	Defines the vertical alignment of the contents within the defined frame (display area).
BackColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	Sets the background color on which to display the text characters (functionality depends on BackStyle property).
BackStyle	Opaque, Transparent	Defines the style of the background on which the text is displayed.
BorderColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	Sets the color for of the defined frame.
BorderWidth	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Sets the width of the border of the defined frame.
ClearAfter	Yes, No	Controls the clearing of the defined display area after the object's termination condition is met.
Data Logging	None, Standard, Response Only, Time Audit Only, Custom	Determines whether data is logged to the data file, and which attributes are included.
Duration	-1 (forever), Numeric values (default in msec), Attribute references.	Determines the duration of the object in milliseconds.
FontBold	Yes, No, Attribute references.	Indicates whether the font should be in bold style.
FontItalic	Yes, No, Attribute references.	Indicates whether the font should be italicized.
FontName	Dropdown list created from those available in the system, string values (e.g., "Courier New").	Indicates the name of the font to be used to display the text.
FontSize	Integer values (e.g., 10, 12)	Indicates the point size of the font.



E-Prime Reference Guide
Chapter 1: E-Studio

Property	Possible Values	Description
FontStrikeout	Yes, No, Attribute references.	Indicates whether the font should be in Strikeout style (a line drawn horizontally through the text).
FontUnderline	Yes, No, Attribute references.	Indicates whether the font should be underlined.
ForeColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	Sets the foreground color for the display object.
Height	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Defines the height of the rectangle designating the text display area.
JumpLabel	String values, Attribute references.	Identifies a Label object within the current Procedure to which program execution jumps when input is received on a mask that has an End Action of "Jump".
Notes	String values	User-entered text useful for storing information or comments related to an object.
OffsetSync	(none), Vertical blank	Sets the object or event with which the offset of the object is synchronized.
OnsetSync	(none), Vertical blank	Sets the object or event with which the onset of the object is synchronized.
PreRelease	Numeric values (default in msec), Attribute references.	Amount of time released during the processing of the current object to allow for setup of the next object.
Tag	String values, Attribute references.	Defines a string value to be logged with the object.
Text	String values, Attribute references.	Sets the Text to be displayed when the object is run.
Timing Mode	Event, Cumulative, Custom	Determines whether timing occurs in Event, Cumulative, or Custom mode.
Width	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10% of full screen window specified for the display), Attribute references.	Defines the width of the rectangle designating the text display area.
WordWrap	Yes, No	Indicates whether or not to wrap the text within the defined frame.
X	Left, Center, Right, Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Sets the X-coordinate position for the anchor position of the display area.
XAlign	Left, Center, Right	Sets the alignment of the defined frame area along the horizontal (x) axis relative to the anchor position (X and Y property values).
Y	Positions (Top, Center, Bottom), Numeric string values (e.g., "10"), Relative values (e.g., "10%"), Attribute references.	Sets the Y-coordinate position for the anchor of the display area.
YAlign	Top, Center, Bottom	Sets the alignment of the y-axis of the defined frame area in relation to the anchor position (X and Y property settings).



1.3.6 *ImageDisplay Object*



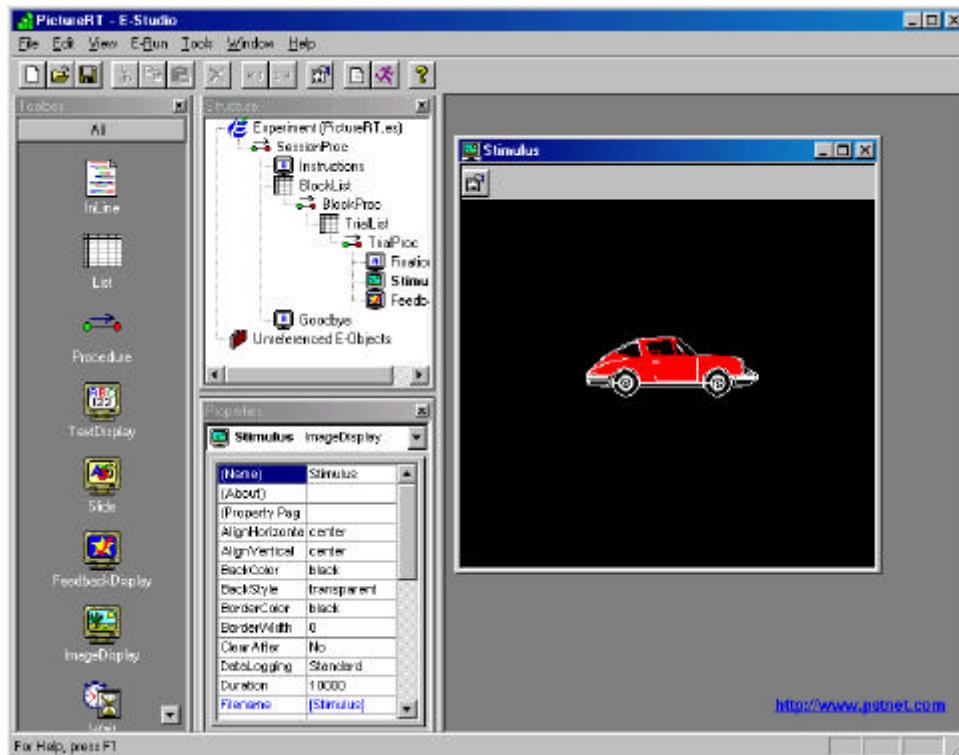
The *ImageDisplay* object as it appears in the *Toolbox*.

1.3.6.1 Overview

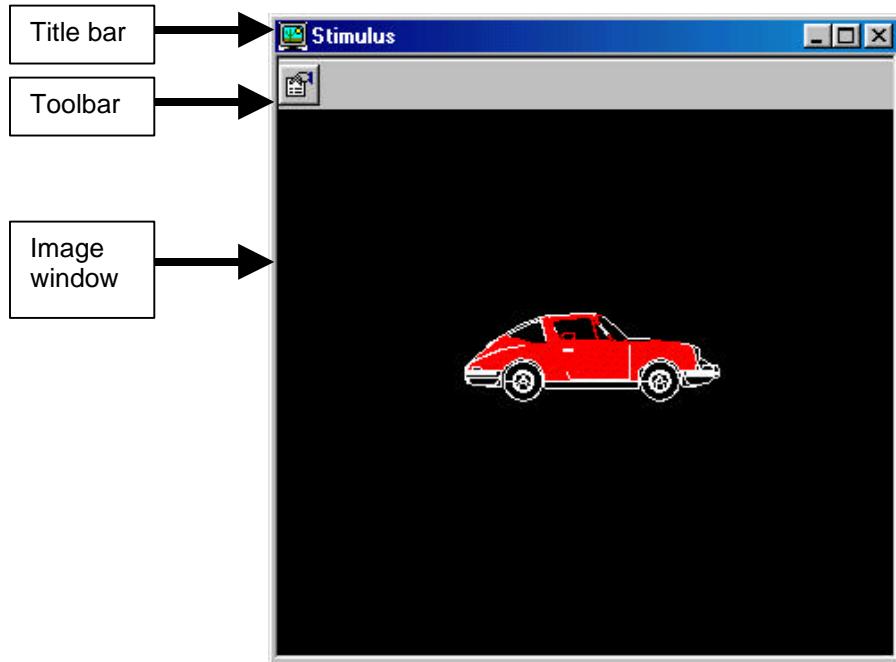
The *ImageDisplay* object is used to display pictures to the subject. Images must be saved in BMP format. Other file formats are not supported and must be converted to BMP. The *ImageDisplay* is used to display a single picture at a time within the defined display area. The *Slide* object should be used to display multiple pictures simultaneously, or to display a combination of pictures and text.

1.3.6.2 Interface

A new *ImageDisplay* object may be opened by clicking the *ImageDisplay* icon in the *Toolbox*, and dragging the object to the *Workspace*. For existing *ImageDisplay* objects, clicking the object once with the mouse in the *Structure* view to select it will result in the display of that object's properties in the *Properties* window. Double clicking an existing *ImageDisplay* in the *Structure* view will result in that object being opened in the *Workspace*, as well as the properties for that object being displayed in the *Properties* window.



An *ImageDisplay* opened in the *Workspace* displays the name of the object, along with the *ImageDisplay* icon in the title bar at the top of the object window. Below the title bar, the toolbar lists the tool buttons associated with the *ImageDisplay*. Currently, the only active tool button is the *Property Pages* button. The *Image* window appears at the bottom of the *ImageDisplay*. The *Image* window displays the contents of the designated image file.



The ImageDisplay object is meant to simulate the display the subject will see during a run of the experiment. Using the Properties window or the Property pages, the properties of the display may be set. The setting of observable properties (i.e., BackColor, Alignment, etc.) will take effect immediately in the window of the ImageDisplay.

The filename for the picture may be indicated directly in the ImageDisplay properties, or the bracket/attribute notation may be used to vary the picture to be displayed. The bracket/attribute notation indicates to E-Run that the information requested varies at run time, and E-Run will look to the context to resolve the value of the attribute name in brackets.



E-Studio will automatically use relative paths. If no path is specified, E-Run will look for files in the current directory, and an error will occur if the files cannot be located. To specify a filename



outside of the current directory, a specific syntax must be used. E-Basic recognizes the backslash ("\") character often used for path definition as an escape character. This character indicates to E-Basic that the first character following the backslash should be read as a special character. For example, within E-Basic, "\t" is read as a tab, while "\n" inserts a new line. To define folders within a path definition, use the forward slash (Figure 1 below) or a double backslash (Figure 2). The double backslash allows E-Basic to read the backslash character as itself.

Stimulus
c:\My Pictures\RedCar.bmp
c:\My Pictures\BlueCar.bmp

Figure 1. Path specification using forward slash.

Stimulus
c:\\My Pictures\\RedCar.bmp
c:\\My Pictures\\BlueCar.bmp

Figure 2. Path specification using double backslash.

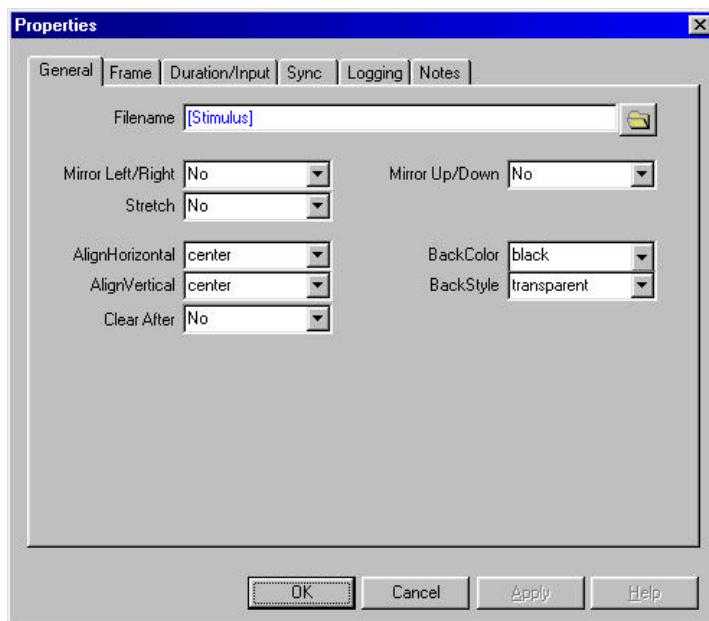
1.3.6.3 Property Pages

The properties for the ImageDisplay may be set using the Property pages. The Property pages may be displayed using the Property Pages tool button on the ImageDisplay toolbar, or by clicking the ellipsis (...) in the Property Pages field in the Properties window.



General Tab

The Property pages open by default to display the General tab. The General tab allows the specification of the picture to be displayed as well as the observable properties of that object (e.g., background color if the image does not fill the defined display area).





Property	Possible Values	Description
Filename	Strings, Attribute references	Sets the name of the image to display. Note that variable and constant information can be concatenated (e.g., [filename].bmp, where [filename] is an attribute defined in a List object).
MirrorLeftRight	Yes, No, Attribute references.	Designates whether the contents of a picture file should be flipped (i.e., turned over) along the horizontal axis.
MirrorUp/Down	Yes, No, Attribute references.	Designates whether the contents of a picture file should be flipped (i.e., turned over) along the vertical axis.
Stretch	Yes, No, Attribute references.	Designates whether the image should be stretched or reduced to fit the defined display area.
AlignHorizontal	Left, Center, Right, Attribute references	Determines the horizontal alignment of the image within the defined display area.
AlignVertical	Top, Center, Bottom, Attribute references	Determines the vertical alignment of the image within the defined display area.
ClearAfter	Yes, No	Controls the clearing of the defined display area after the object's termination condition is met.
BackColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	The BackColor field allows the user to designate the color for the background in the ImageDisplay. If the picture does not fill the entire display area, the BackColor will fill in the remaining area.
BackStyle	Opaque, Transparent	Designates the style of the background. In most cases, if using a common display area, the opaque backstyle will result in the removal of the previous display. A transparent backstyle results in an overlay of the current image over the previous display.

1.3.6.4 Properties Window

The Properties of the ImageDisplay may be set using the fields in the Properties window.

Property	Possible Values	Description
(Name)	String values	Assigns a user-entered string as the name of the ImageDisplay.
(About)	-----	Displays the About ImageDisplay dialog.
(Property Pages)	-----	Opens the Property pages for the ImageDisplay.
AlignHorizontal	Left, Center, Right, Attribute references	Defines the horizontal alignment of the contents within the defined frame (display area).
AlignVertical	Top, Center, Bottom, Attribute references	Defines the vertical alignment of the contents within the defined frame (display area).
BackColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	The background color on which to display the picture (i.e., if the picture does not take up the full area, the area behind the picture will be filled with the BackColor setting). Functionality depends on BackStyle property.
BackStyle	Opaque, Transparent, Attribute references	Sets the style of the background for the display object.
BorderColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	Sets the color of the border of the defined display area.
BorderWidth	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Reflects the size of the border of the display area.
ClearAfter	Yes, No	Controls the clearing of the defined display area after the object's termination condition is met.



ImageDisplay Properties Continued...

Property	Possible Values	Description
DataLogging	None, Standard, Response Only, Time Audit Only, Custom	Determines the category of variables logged for the object.
Duration	-1 (forever), Numeric values (default in msec), Attribute references.	Determines the duration of the object in milliseconds.
Filename	String values (e.g., Image.bmp), Attribute references.	Designates the name of the file to load for display.
Height	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Sets the height of the display area.
JumpLabel	String values, Attribute references.	Identifies a Label object within the current Procedure to which program execution jumps when input is received on a mask that has an End Action of "Jump".
MirrorLeftRight	Yes, No, Attribute references.	Designates whether or not to flip the image horizontally.
MirrorUpDown	Yes, No, Attribute references.	Designates whether or not to flip the image vertically.
Notes	String values	User-entered text useful for storing information or comments related to an object.
OffsetSync	(none), Vertical blank	Sets the object or event with which the offset of the object is synchronized.
OnsetSync	(none), Vertical blank	Sets the object or event with which the onset of the object is synchronized.
PreRelease	Numeric values (default in msec), Attribute references.	Amount of time released during the processing of the current object to allow for setup of the next object.
Stretch	Yes, No	Designates whether or not the picture should be expanded or reduced to fit the display area.
Tag	String values, Attribute references.	Defines a string value to be logged with the object.
TimingMode	Event, Cumulative, Custom	Sets the timing method to be used by the object. Cumulative timing is used to absorb the processing time between events, and to maintain a certain interval duration between events or between trials.
Width	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Sets the width of the display area.
X	Left, Center, Right, Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Sets the X-coordinate position for the anchor position of the display area.
XAlign	Left, Center, Right	Sets the alignment of the defined frame area along the horizontal (x) axis relative to the anchor position (X and Y property values).
Y	Top, Center, Bottom, Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Sets the Y-coordinate position for the anchor of the display area.
YAlign	Top, Center, Bottom	Sets the alignment of the y-axis of the defined frame area in relation to the anchor position (X and Y property settings).



1.3.7 Slide Object



The Slide object as it appears in the Toolbox.

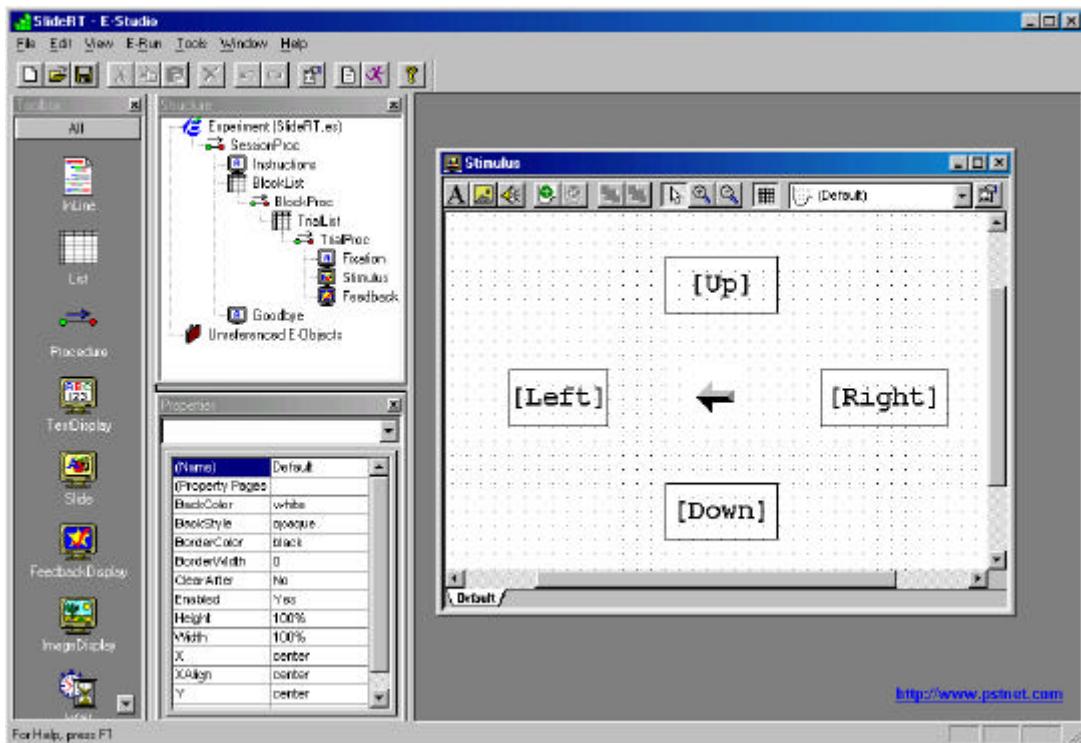
1.3.7.1 Overview

The Slide object is used to present multiple visual stimuli (e.g., text, multiple images, text and images, etc.), or combinations of text, images, and sound concurrently. The Slide object holds a collection of SlideState objects, which are used to organize the presentation of stimuli. Individual stimuli composing a single presentation event are entered on SlideStates using sub-objects. This hierarchy of objects permits tremendous flexibility in the presentation of stimuli.

Properties are set independently for the Slide object, each of its child SlideState objects, and their sub-objects. The parent object (i.e., Slide) maintains parameters relevant to the duration of the object, the collection of input, and termination conditions.

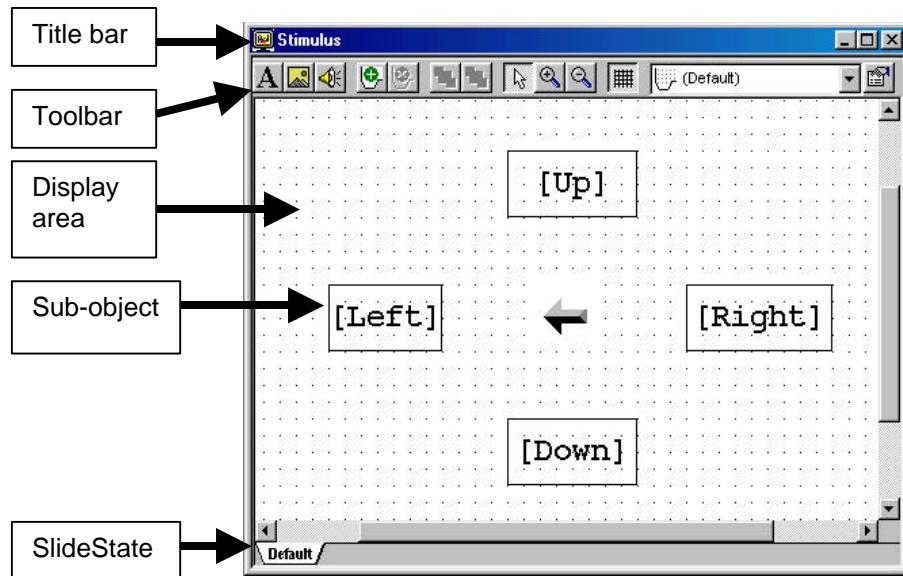
1.3.7.2 Interface

A new Slide object may be created by clicking the Slide icon in the Toolbox, and dragging the object to the Workspace. For existing Slide objects, clicking the object once with the mouse in the Structure view to select it will result in the display of that object's properties in the Properties window. Double clicking an existing Slide object in the Structure view will open the Slide in the Workspace, and the properties for that object will be displayed in the Properties window.





A Slide object opened in the Workspace displays the name of the object and the Slide object icon in the title bar at the top of the object window. Below the title bar, the toolbar lists the tool buttons associated with the Slide object.



Tool Button	Description
	The SlideText button allows the placement of a SlideText sub-object on the Slide object to define an area in which to display text.
	The SlideImage button allows the placement of a SlideImage sub-object on the Slide to define an area in which to display a bitmap.
	The SlideSoundOut button allows the placement of a SlideSoundOut sub-object on the Slide in which to present a WAV file.
	The Add SlideState button permits the addition of a SlideState object to the collection.
	The Remove SlideState button removes the selected SlideState object from the collection.
	The Bring To Front button places the selected sub-object in the foreground if several sub-objects are overlapping.
	The Send To Back button sends the selected sub-object to the background if several sub-objects are overlapping.
	The Select Objects button activates the cursor so that specific sub-objects may be selected. When selected, the borders of the selected objects are defined by Windows resizing handles.
	The Zoom In button allows the user to zoom in on a selected portion of the Slide object.
	The Zoom Out button allows the user to zoom out on a selected portion of the Slide object.
	The Grid button toggles the display of the grid in the object's display area.
	Allows selection of a specific item (i.e., the parent Slide object, SlideState, or sub-object).
	The Properties button displays the Property pages for the display area of the Slide object, or the selected sub-object.



The display area appearing at the bottom of the Slide object window is used to enter and organize the text and images to be displayed at run-time. The display area is, itself, a SlideState object. By default, a new Slide object is created with a single SlideState object. Additional SlideState objects may be added (up to a maximum of 10). The Slide object is the parent to the collection of SlideState objects. In turn, a SlideState object is the parent to a collection of SlideStim objects, also called sub-objects, which are used to enter a single text, image, or sound stimulus. Combinations of sub-objects compose the structure of the SlideState. Multiple SlideState objects allow the customization of varying stimuli to be presented by the same parent object. Refer to section 1.3.7.5 for a complete description of SlideState objects.

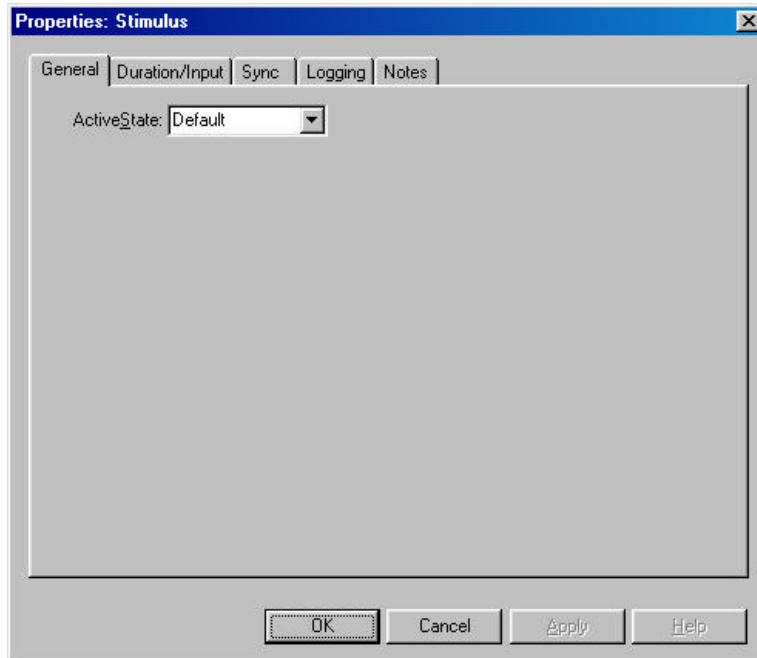
1.3.7.3 Property Pages

The properties for the Slide object may be set using the Property pages. The Property pages may be displayed using the Property Pages tool button on the Slide object toolbar.



General Tab

The General tab displays the ActiveState, indicating the SlideState object to be displayed when the Run method is called for the parent object. The value for ActiveState may be the name of a specific SlideState, or may be an attribute reference, in order to vary the SlideState presentation (e.g., per condition).



Property	Possible Values	Description
ActiveState	String values representing names of SlideState objects, Attribute references	Identifies the SlideState to be displayed when Run is called for the Slide object.



1.3.7.4 Properties Window

The Properties for the Slide object may be set using the fields in the Properties window. Properties include information concerning the observable features of the Slide (e.g., BackColor), as well as parameters related to the duration and termination of the object, and acceptable input.

Property	Possible Values	Description
(Name)	String values	Assigns a user-entered string as the name of the Slide.
(About)	-----	Displays the About Slide dialog.
(Property Pages)	-----	Reveals the property page interface to specify the properties for the selected object.
ActiveState	Name of SlideState, Attribute references	Identifies the SlideState to be displayed when the Run method of the parent object is called.
DataLogging	None, Standard, Response Only, Time Audit Only, Custom	Determines the category of variables logged for the object.
Duration	-1 (forever), Numeric values (default in msec), Attribute references.	Determines the duration of the object in milliseconds.
JumpLabel	String values.	Identifies a Label object within the current Procedure to which program execution jumps when input is received on a mask that has an End Action of "Jump".
Notes	String values	User-entered text useful for storing information or comments related to an object.
OffsetSync	(none), Vertical blank	Sets the object or event with which the offset of the object is synchronized.
OnsetSync	(none), Vertical blank	Sets the object or event with which the onset of the object is synchronized.
PreRelease	Numeric values (in msec), Attribute references.	Amount of time released during the processing of the current object to allow for setup of the next object.
Tag	String values, Attribute references.	Defines a string value to be logged with the object.
TimingMode	Event, Cumulative, Custom	Sets the timing method to be used by the object. Cumulative timing is used to absorb the processing time between events, and to maintain a certain interval duration between events or between trials.

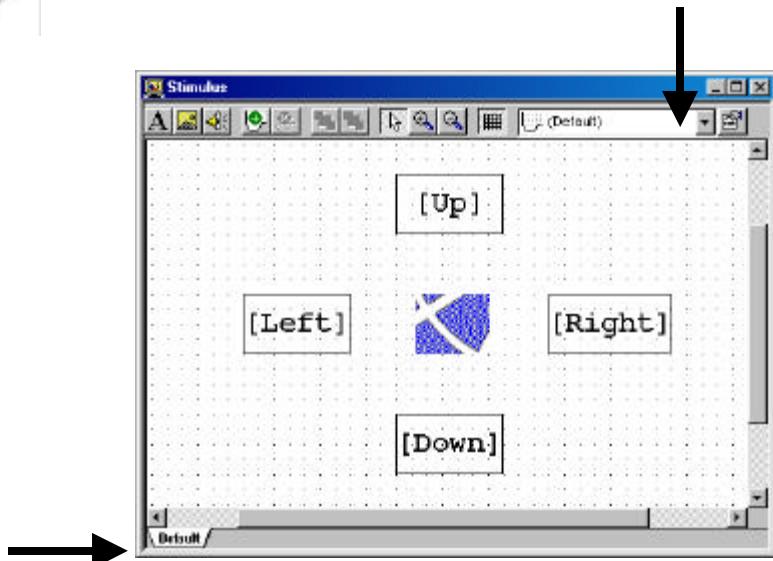
1.3.7.5 SlideState Object

Overview

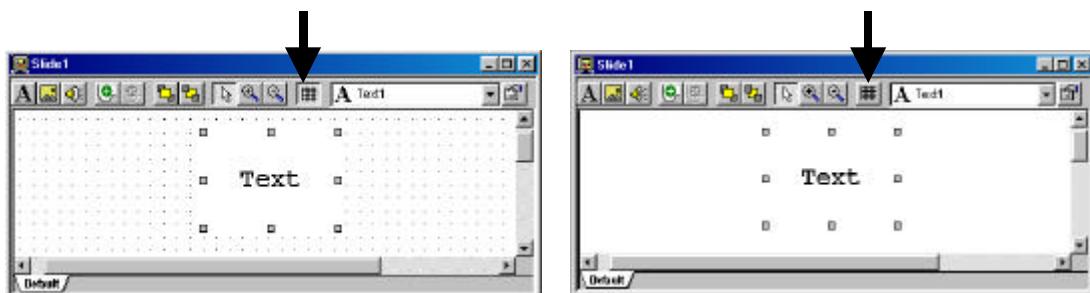
The Slide object is parent to a collection of SlideState objects. SlideState objects are used to enter and organize the stimuli to be presented at run-time. The collection of SlideStates maintained by the Slide allows customization for specific conditions independently.

Interface

SlideState objects are accessed by clicking the tabs located at the bottom of the display window, or through the dropdown box on the Slide toolbar.



By default, the display area (i.e., SlideState object) is marked by dots forming a grid. This grid is provided to aid in the placement of sub-objects. The grid may be displayed or hidden by toggling the Grid button in the toolbar.



The display area is meant to organize and simulate the display the subject will see during a run of the experiment (minus the grid). Sub-objects are used to define specific locations in the display area in which stimuli (e.g., text, images, sound files) may be placed. To enter text, image, or sound stimuli in the display area, use the sub-object buttons on the Slide toolbar. After selection of a button, the cursor will be modified to include a picture of the type of sub-object being entered (i.e., the arrow cursor will be accompanied by a picture of the button which is selected).

Tool Button	Description	Cursor
	Inserts a SlideText sub-object.	
	Inserts a SlideImage sub-object.	
	Inserts a SlideSound sub-object.	

While the appropriate cursor is displayed, click in the grid area to enter the sub-object. A sub-object of a default size will be created in the location of the click. When a sub-object is created, the area of the sub-object will be defined by resizing handles. The sub-object may be resized by clicking and dragging the resizing handles until the object is of the desired size (Figure 1 below). A sub-object may be re-positioned by placing the cursor over the sub-object until the cursor changes to a four-sided arrow (Figure 2), clicking the sub-object, and dragging the sub-object to the desired location.

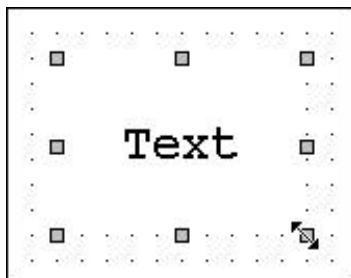


Figure 1. Resizing a sub-object

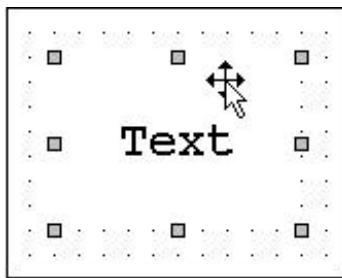


Figure 2. Moving a sub-object

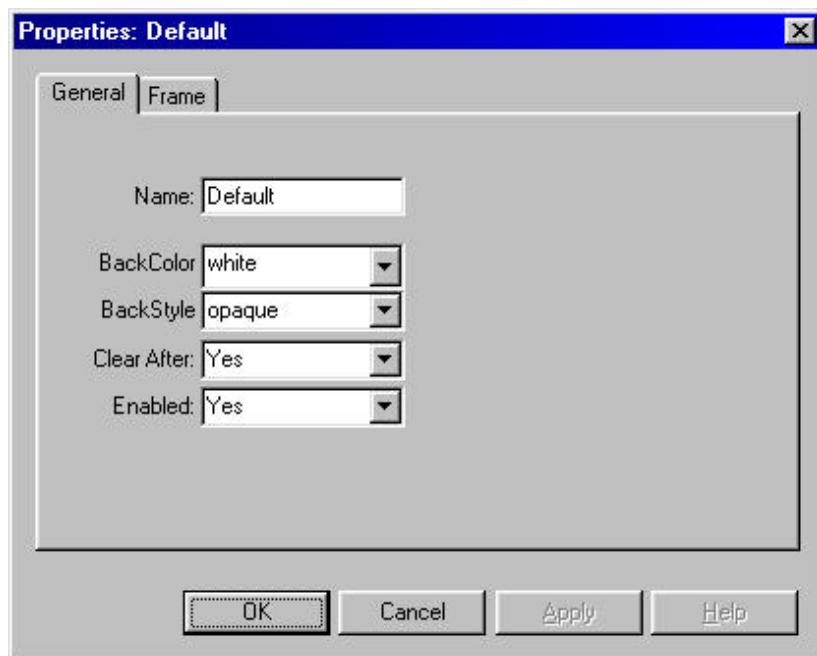
Alternatively, the sub-object's size and placement may be set using the properties specific to the sub-object. To display a sub-object's properties, select the sub-object and click the Properties button, or view the properties via the Properties window. Refer to the Sub-Objects section in this chapter (section 1.3.7.7) for a complete description of sub-object types and properties.

Property pages

The Property pages for a particular SlideState object are accessed by selecting the SlideState and clicking the Property Pages button. The SlideState Property pages permit the user to set properties related to the size and appearance of the each SlideState object. Because each SlideState maintains its own property settings, features such as backgrounds, fonts, and borders may be customized independently for each SlideState object, and for the sub-objects housed within the SlideState.

General Tab

The General tab specifies the settings for the name of the SlideState, the background color and style, and settings related to the enabling and cleanup action of the SlideState.





Property	Possible Values	Description
Name	String values	Assigns a user-entered string as the name of the SlideState.
BackColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	Sets the background color for the SlideState object.
BackStyle	Opaque, Transparent	Sets the style of the background for the SlideState object.
Clear After	Yes, No	Controls the clearing of the defined display area after the object's termination condition is met.
Enabled	Yes, No	Determines whether or not the SlideState is enabled (i.e., available).

Properties Window

Property	Possible Values	Description
(Name)	String values	Assigns a user-entered string as the name of the SlideState object.
(Property Pages)		Reveals the property page interface to specify the properties for the selected object.
BackColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	Sets the background color for the object.
BackStyle	Opaque, Transparent	Sets the style of the background for the object.
BorderColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	Sets the color of the border to be drawn around the display area.
BorderWidth	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	The width of the border displayed around the outside of the defined frame.
Clear After	Yes, No	Controls the clearing of the defined display area after the object's termination condition is met.
Enabled	Yes, No	Does not display text, but still utilizes the duration with a blank screen.
Height	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Defines the height of the rectangle designating the display area.
Width	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Defines the width of the rectangle designating the Slide area.
X	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	The X coordinate/position of the SlideState.
XAlign	Left, Center, Right	The alignment of the frame relative to the X coordinate.
Y	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	The Y coordinate/position of the SlideState.
YAlign	Top, Center, Bottom	The alignment of the frame relative to the Y coordinate.

1.3.7.6 Sub-Objects

Each SlideState object within the Slide object collection houses various sub-objects used to present text, images or sounds. A SlideState may contain a single sub-object, or a combination of several sub-objects to present text, graphics and sounds concurrently. Different types of sub-objects are available to allow the presentation of text, images and sounds, and set parameters specific to these different types of stimuli.



SlideText Sub-Object

The SlideText sub-object is used to place text within a SlideState object. The text may be typed directly in the Text area defined by the SlideText sub-object, or in the Text property field in the Properties window. Click the SlideText sub-object once to display the properties in the Properties window. Click the sub-object twice to enter edit mode and type directly into the sub-object's Text area.

Property pages

The Property pages for the SlideText sub-object are identical to those specific to the TextDisplay object (i.e., General, Frame, Font). Refer to section 1.3.5.3 for a description of those pages.

Properties Window

The Properties for the SlideText sub-object may be set using the Properties window. Click the SlideText sub-object to select it. Resizing handles on the perimeter of the sub-object indicate that it has been selected, and the Properties window will display the properties for the selected sub-object.

Property	Possible Values	Description
(Name)	String values	Assigns a user-entered string as the name of the sub-object.
(Property Pages)		Reveals the property page interface to specify the properties for the selected object.
AlignHorizontal	Left, Center, Right, Attribute references	Defines the horizontal alignment of the contents within the frame (display area).
AlignVertical	Top, Center, Bottom, Attribute references	Defines the vertical alignment of the contents within the defined frame (display area).
BackColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	The background color on which to display the text characters (functionality depends on BackStyle property).
BackStyle	Opaque, Transparent	Defines the style of the background on which the text is displayed.
BorderColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	The color used to display a border around the edge of the defined frame.
BorderWidth	Numeric values in pixels (e.g., "10"), Attribute references.	The width of the border displayed around the edge of the defined frame.
FontBold	Yes, No, Attribute references.	Indicates whether the font should be in bold style.
FontItalic	Yes, No, Attribute references.	Indicates whether the font should be italicized.
FontName	Dropdown list created from those available in the system, string values (e.g., "Courier New").	Indicates the name of the font to be used to display the text.
FontSize	Numeric values (e.g., 10, 12).	Indicates the point size of the font.



Property	Possible Values	Description
FontStrikeout	Yes, No, Attribute references.	Indicates whether the font should be in Strikeout style (a line drawn horizontally through the text).
FontUnderline	Yes, No, Attribute references.	Indicates whether the font should be underlined.
ForeColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	Sets the foreground color for the display object.
Height	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Defines the height of the rectangle designating the text display area.
Text	String values, Attribute references.	Sets the Text to be displayed when the object is run.
Width	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10% of full screen window specified for the display), Attribute references.	Defines the width of the rectangle designating the text display area.
WordWrap	Yes, No	Indicates whether or not to wrap the text within the defined frame.
X	Left, Center, Right, Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Sets the X-coordinate position for the anchor position of the display area.
XAlign	Left, Center, Right	Sets the alignment of the defined frame area along the horizontal (x) axis relative to the anchor position (X and Y property values).
Y	Top, Center, Bottom, Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Sets the Y-coordinate position for the anchor of the display area.
YAlign	Top, Center, Bottom	Sets the alignment of the y-axis of the defined frame area in relation to the anchor position (X and Y property settings).

SlideImage Sub-Object

The SlideImage sub-object is used to place an image in bitmap format within the SlideState object. The name of the bitmap may be typed directly in the Filename field in the sub-object properties, or the image may be varied using an attribute reference (e.g., [ImageName]) in the Filename field. If the actual filename is entered, the bitmap will be displayed in the sub-object. If an attribute reference is entered in the Filename field, a placeholder bitmap is displayed, since the actual file to be displayed would not be determined until run-time.

Property pages

The Property pages for the SlideImage sub-object are identical to those specific to the ImageDisplay object (i.e., General, Frame). Refer to section 1.3.6.3 for a description of those pages.



Properties Window

When the sub-object is selected, properties for the SlidelImage sub-object may be set using the Properties window.

Property	Possible Values	Description
(Name)	String values.	Assigns a user-entered string as the name of the sub-object.
AlignHorizontal	Left, Center, Right, Attribute references	Determines the horizontal alignment of the image within the defined display area.
AlignVertical	Top, Center, Bottom, Attribute references	Determines the vertical alignment of the image within the defined display area.
BackColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	The BackColor field allows the user to designate the color for the background in the sub-object. If the picture does not fill the entire display area, the BackColor will fill in the remaining area.
BackStyle	Opaque, Transparent	Designates the style of the background. In most cases, using a common display area, the opaque backstyle will result in the removal of the previous display. A transparent backstyle results in an overlay of the current image over the previous display.
BorderColor	Color value (RGB values), Specific color strings (e.g., black, maroon, lime), Attribute references.	Sets the color of the border drawn around the edge of the display area.
BorderWidth	Numeric values in pixels (e.g., "10"), Attribute references.	Sets the size of the border drawn around the edge of the display area.
Filename	String values, Attribute references.	Designates the name of the image file to display.
Height	Numeric values in pixels (e.g., "10"), Relative values (e.g., "10%").	Defines the height of the rectangle designating the display area.
MirrorLeftRight	Yes, No, Attribute references.	Designates whether the contents of a picture file should be flipped (i.e., turned over) along the horizontal axis.
MirrorUpDown	Yes, No, Attribute references.	Designates whether the contents of a picture file should be flipped (i.e., turned over) along the vertical axis.
Stretch	Yes, No, Attribute references.	Designates whether the image should be stretched or reduced to fit the display area.
Width	Numeric values in pixels (e.g., "10"), Relative values (e.g., "10%").	Defines the width of the rectangle designating the display area.
X	Positions (Left, Center Right), Numeric string values (e.g., "10"), Relative values (e.g., "10%"), Attribute references.	Sets the anchor position along the horizontal axis to a percentage of the x resolution, or to a fixed pixel location.
XAlign	Left, Center, Right	Determines the horizontal positioning of the defined frame in relation to the X and Y anchor position.
Y	Positions (Top, Center, Bottom), Numeric string values (e.g., "10"), Relative values (e.g., "10%"), Attribute references.	Sets the anchor position along the vertical axis to a percentage of the y resolution, or to a fixed pixel location.
YAlign	Top, Center, Bottom	Determines the vertical positioning of the defined frame in relation to the X and Y anchor position.



SlideSoundOut Sub-Object

The SlideSoundOut sub-object is used to place an audio stimulus within a SlideState object. The name of the WAV file may be typed in directly in the Filename field in the SlideSoundOut properties, or the WAV file may be varied by using an attribute reference (e.g., [SoundFileName]) in the Filename field. Within E-Studio, a SlideSoundOut sub-object is viewable so that it may be selected and its properties may be set. At run-time, the SlideSoundOut sub-object is not visible, nor does it affect the display presented by the Slide object.

Property pages

The Property pages for the SlideSoundOut sub-object are identical to those specific to the SoundOut object (i.e., General). Refer to section 1.3.10.3 for a description of those pages.

Properties Window

Properties for the SlideSoundOut sub-object may be set using the Properties window. Click the sub-object to select it. Resizing handles on the perimeter of the sub-object indicate that it has been selected, and the Properties window will display the properties for the selected sub-object.

Property	Possible Values	Description
(Name)	String values.	Assigns a user-entered string as the name of the sub-object.
(Property Pages)		Reveals the property page interface to specify the properties for the selected object.
EndSoundAction	(none), Terminate, Jump	Determines the action to be taken upon termination of the playback.
Filename	Strings, Attribute references	Determines the audio file to be loaded by the sub-object.
Height	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Ignored. The SlideSoundOut sub-object is visible only at design-time, and the size of the sub-object is not modifiable.
Loop	Yes, No, Attribute references.	Sets playback to loop continuously between the StartOffset and StopOffset property settings.
MaxLength	Integer values	Determines the maximum audio buffer size.
Pan	-1000 to 10000	Sets the panning level for playback.
Pan Control	Yes, No	Enables or disables panning control for playback.
StartOffset	Integer values, Attribute references	Determines the offset (in bytes) from which audio playback will begin.
StopOffset	Integer values, Attribute references	Determines the offset (in bytes) at which audio playback will terminate.
StopAfter	Yes, No	Determines whether the playback will continue after the termination of the Slide object's Run method.
Volume Control	Yes, No	Enables or disables control of the volume level for playback.
Volume	-10000 to 0	Sets the volume level for playback.
Width	Numeric values in pixels (e.g., "10"), Relative values (e.g., 10%), Attribute references.	Ignored. The SlideSoundOut sub-object is visible only at design-time, and the size of the sub-object is not modifiable.
X	Positions (Top, Center, Bottom), Numeric string values (e.g., "10"), Relative values (e.g., "10%"), Attribute references.	Sets the X-coordinate position for the sub-object. Within E-Studio, the SlideSoundOut sub-object may be positioned as a convenience. It is not visible at run-time.
Y	Positions (Top, Center, Bottom), Numeric string values (e.g., "10"), Relative values (e.g., "10%"), Attribute references.	Sets the Y-coordinate position for the sub-object. Within E-Studio, the SlideSoundOut sub-object may be positioned as a convenience. It is not visible at run-time.



1.3.8 FeedbackDisplay Object



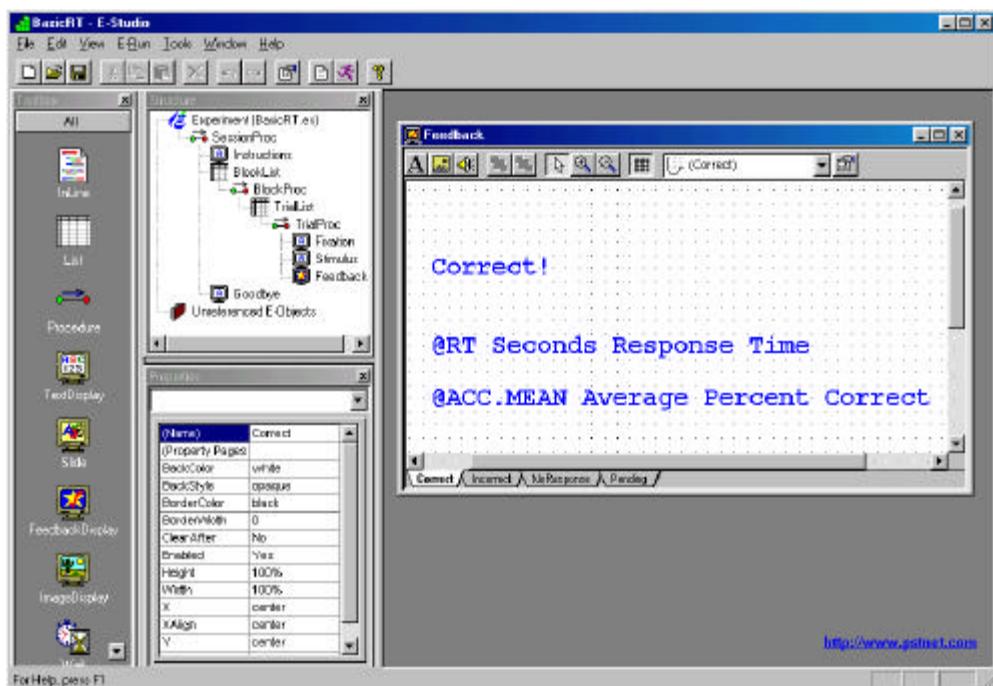
The FeedbackDisplay object as it appears in the Toolbox.

1.3.8.1 Overview

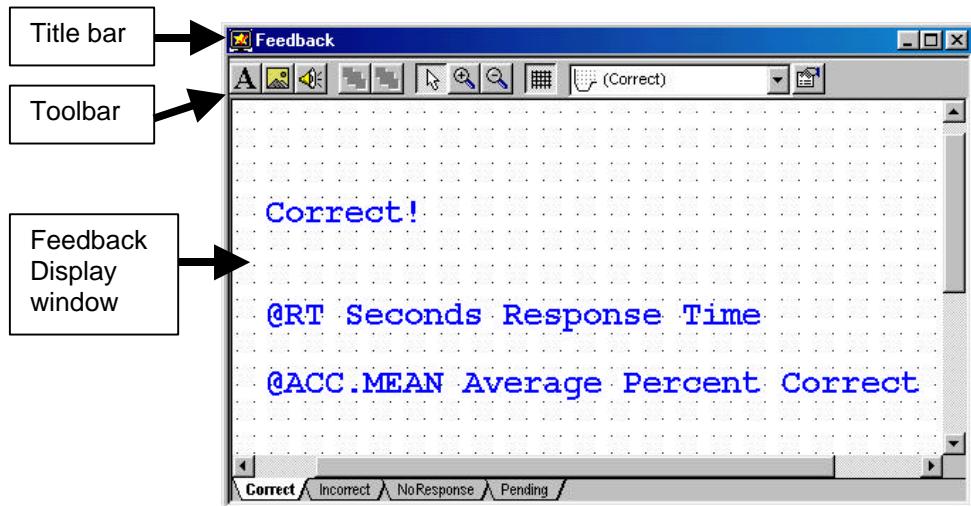
The FeedbackDisplay object is derived from the Slide object, but is specifically designed for the presentation of feedback. Because the FeedbackDisplay object is itself a kind of Slide object, it houses a collection of SlideState objects each with its own sub-objects and property settings. Thus, a single FeedbackDisplay object may be customized to present varying types of feedback. However, the FeedbackDisplay object is unique in that it is used to present feedback based on the input collected by another object. Thus, the FeedbackDisplay object interacts with another object to obtain the input information, then evaluates the input and presents the appropriate feedback information.

1.3.8.2 Interface

A new FeedbackDisplay may be opened by clicking the FeedbackDisplay object icon in the Toolbox, and dragging the object to the Workspace or to a location within the hierarchy represented in the Structure view. For an existing FeedbackDisplay object, clicking the object once with the mouse in the Structure view to select it will result in the display of that object's properties in the Properties window. Double clicking an existing FeedbackDisplay in the Structure view will open the object in the Workspace, and the properties for that object will be displayed in the Properties window.



A FeedbackDisplay object opened in the Workspace displays the name of the object and the FeedbackDisplay icon in the title bar at the top of the object window. Below the title bar, the toolbar lists the tool buttons associated with the FeedbackDisplay object.

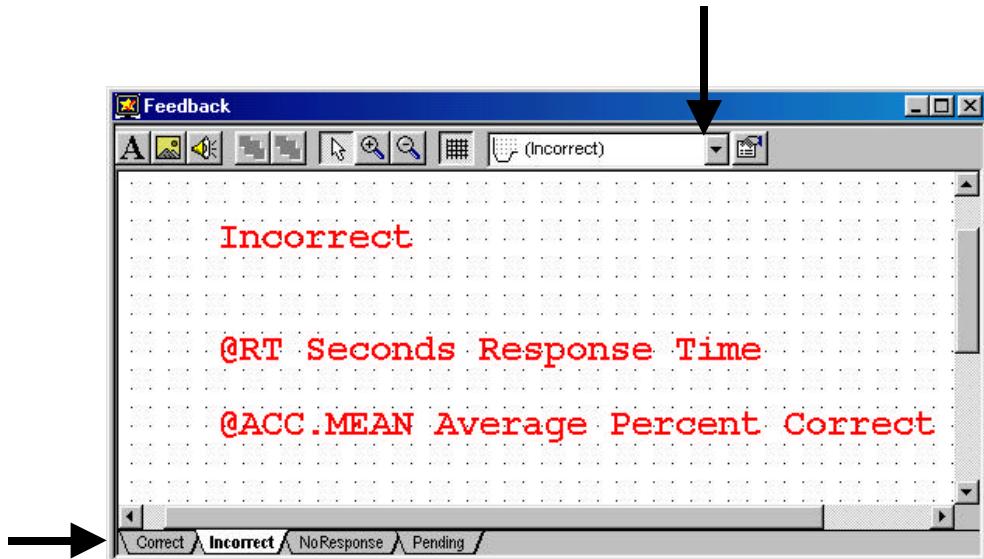


Tool Button	Description
	Allows placement of a SlideText sub-object for the display of text.
	Allows placement of a SlideImage sub-object for displaying an image.
	Allows a WAV file to be presented with the feedback.
	Brings the selected sub-object to the foreground if several sub-objects are overlapping.
	Sends the selected sub-object to the background if several sub-objects are overlapping.
	Activates the cursor in order to select specific sub-object.
	Allows the user to zoom in on a selected portion of the FeedbackDisplay object by clicking the left mouse button. Right clicking performs the reverse action (zoom out).
	Allows the user to zoom out on a selected portion of the FeedbackDisplay object by clicking the left mouse button. Right-clicking performs the reverse action (zoom in).
	Toggles the display of the grid in the display area.
	Dropdown box allowing the selection of the item to receive focus (e.g., entire FeedbackDisplay object, SlideState, sub-object) for the display of properties and various alterations.
	Displays the Property pages for the item displayed in the dropdown box (e.g., FeedbackDisplay, SlideState, sub-object).

The display presented by the FeedbackDisplay object is represented in the window below the toolbar. The FeedbackDisplay window contains a set of tabbed pages defining the different types of feedback to be presented during the experiment. Separate tabs are presented in order to define feedback for correct and incorrect responses, as well as for non-responses, and responses that may be pending (this option reserved for future use).



Each tab in the FeedbackDisplay window is actually a SlideState object, and the FeedbackDisplay object is the parent to the collection of SlideStates (refer to section 1.3.7.6 for a complete description of SlideState objects). The FeedbackDisplay object maintains its own properties, properties applied to each SlideState object, and properties applied to each sub-object defined on the SlideState objects. The user must take care to select the appropriate object for modification and the display of properties. Each item contained within a FeedbackDisplay object may be selected by clicking it, or by selecting it from the dropdown box located on the FeedbackDisplay object's toolbar. To select a specific sub-object, the SlideState object containing that sub-object must first be selected. For example, to modify the text presented in relation to an incorrect response, first select the SlideState object named "Incorrect" by clicking the "Incorrect" tab at the bottom of the FeedbackDisplay window, or through the dropdown box.



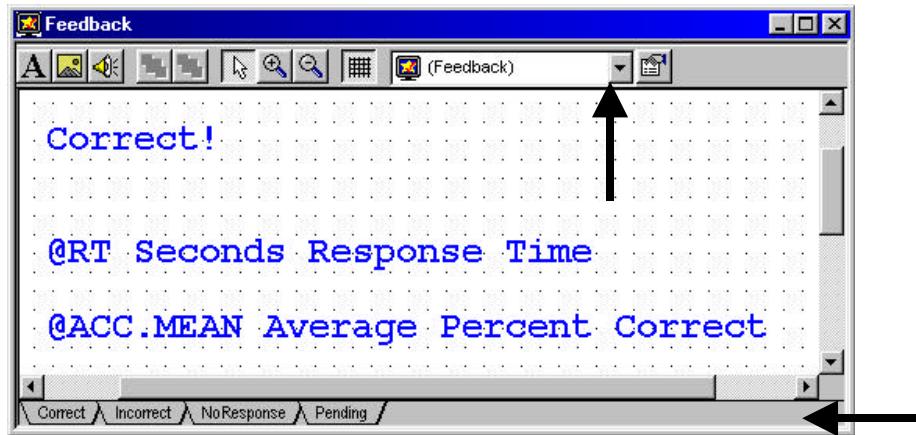
Once the appropriate SlideState object is selected, its sub-objects may be viewed and selected. The FeedbackDisplay object itself is selected by clicking the frame area surrounding the display, or through the dropdown box on the toolbar. The properties displayed in the Properties window are relevant to the currently selected object or sub-object.

1.3.8.3 Property Pages

Property pages for the FeedbackDisplay object may be displayed using the Property Pages tool button on the FeedbackDisplay object toolbar.

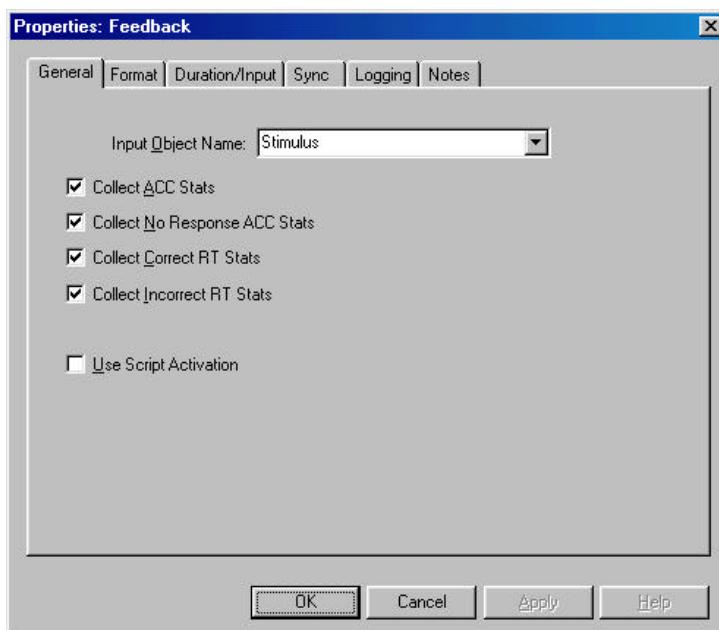


Because the FeedbackDisplay object houses collections of SlideState objects and SlideStim objects (i.e., sub-objects), it is important to note that the properties displayed in the Properties window are in reference to the currently selected object. The user must take care to select the appropriate object in order to display the relevant properties for modification. To select the FeedbackObject, click on the frame area surrounding the display, or through the dropdown box on the toolbar.



General Tab

The General tab identifies the input object for which the feedback information will be determined, and determines the types of statistics to collect.

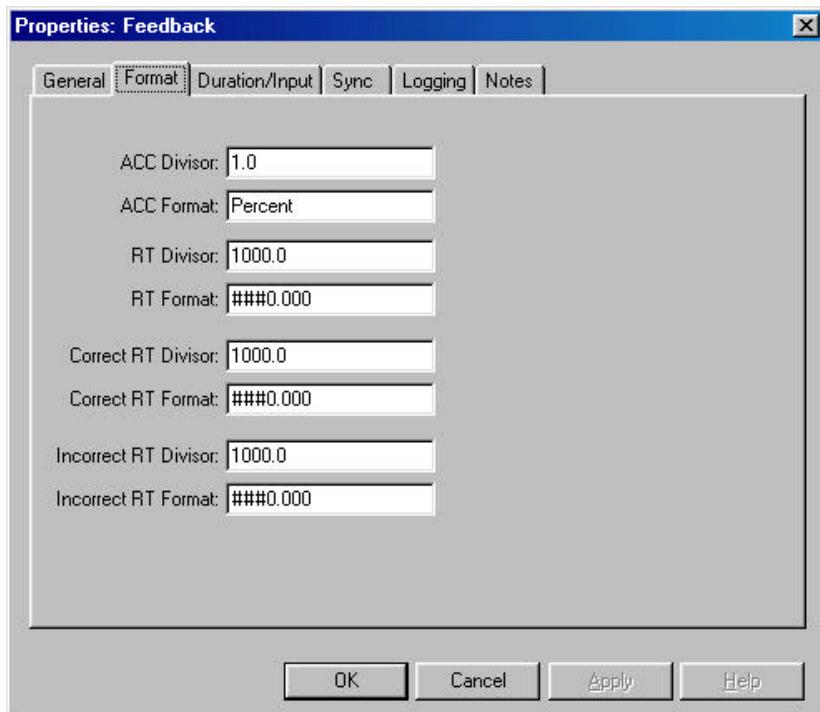


Field	Description
Input Object Name	Identifies the object collecting input for which the feedback will be displayed and statistics will be calculated.
Collect ACC Stats	Determines whether accuracy statistics are collected.
Collect No Response ACC Stats	Determines whether non-responses are included in accuracy statistic calculations.
Collect Correct RT Stats	Determines whether reaction time statistics are collected for correct responses.
Collect Incorrect RT Stats	Determines whether reaction time statistics are collected for incorrect responses.
Use Script Activation	Determines whether the object should use custom user script for its logging activation.



Format Tab

The Format tab defines the format used when displaying feedback. The divisor fields determine the units in which the feedback for accuracy and reaction time occurs. A general format may be applied to all responses, or the format may be customized for correct and incorrect responses.



Field	Possible Values	Description
ACC Divisor	Numeric	Determines the units for the accuracy display.
ACC Format	Percent	Determines the format in which accuracy feedback is displayed.
RT Divisor	Numeric	Determines the units for the reaction time display.
RT Format	Numeric	Determines the number of significant digits for the reaction time display.
Correct RT Divisor	Numeric	Determines the units for the correct reaction time display.
Correct RT Format	Numeric	Determines the number of significant digits for the correct reaction time display.
Incorrect RT Divisor	Numeric	Determines the units for the incorrect reaction time display.
Incorrect RT Format	Numeric	Determines the number of significant digits for the incorrect reaction time display.

1.3.8.4 Properties Window

The Properties of the FeedbackDisplay object may be set using the fields in the Properties window.



E-Prime Reference Guide
Chapter 1: E-Studio

Property	Possible Values	Description
(Name)	Strings	Represents the name or label of the selected object.
(About)	None	Displays the About FeedbackDisplay dialog.
(Property Pages)	None	Reveals the property page interface to specify the properties for the selected object.
ACCDivisor	Real	Determines the units in which the overall accuracy feedback occurs.
ACCFormat	Percent	Determines the format in which the overall accuracy feedback occurs.
CollectACCStats	Yes, No	Determines whether accuracy statistics are collected.
CollectCorrectRTStats	Yes, No	Determines whether reaction time statistics are collected for correct responses independently.
CollectIncorrectRTStats	Yes, No	Determines whether reaction time statistics are collected for incorrect responses independently.
CollectNoRespACCStats	Yes, No	Determines whether non-responses are included in the calculation of accuracy statistics.
CorrectRTDivisor	Real	Divisor used for the retrieval of reaction time statistics for correct responses.
CorrectRTFormat	Real	Format used for the display of RT statistics for correct responses.
DataLogging	None, Standard, Response Only, Time Audit Only, Custom	Indicates the category of variables logged in the data file for the input.
Duration	-1 (forever), Numeric values (default in msec), Attribute references.	Determines the duration of the object in milliseconds.
IncorrectRTDivisor	Real	Divisor used for the retrieval of reaction time statistics for incorrect responses.
IncorrectRTFormat	Real	Format used for the display of RT statistics for incorrect responses.
InputObjectName	Object name	This represents the name of the object collecting the response to be scored. InputObjectName MUST be specified.
JumpLabel	String values	Identifies a Label object within the current Procedure to which program execution jumps when input is received on a mask that has an End Action of "Jump".
Notes	String values	User-entered text useful for storing information or comments related to an object.
OffsetSync	(none), Vertical blank	Determines the object or event with which the offset of the object is synchronized.
OnsetSync	(none), Vertical blank	Determines the object or event with which the onset of the object is synchronized.
PreRelease	Numeric values (default in msec), Attribute references.	Amount of time released during the processing of the current object to allow for setup of the next object.
RTDivisor	Real	Determines the divisor to be used in the retrieval of RT stats.
RTFormat	Real	Determines the format to be used for the display of RT stats.
Tag	String values, Attribute references.	User-defined string value to be logged with the object.
TimingMode	Event, Cumulative, Custom	Sets the timing method to be used by the object. Cumulative timing is used to absorb the processing time between events, and to maintain a certain interval duration between events or between trials.
UseScriptActivation	Yes, No	Determines whether the FeedbackDisplay object should use custom user script for its logging activation.



1.3.8.5 FeedbackDisplay Macros

The following table lists macros that may be used to display various summary statistics in conjunction with a FeedbackDisplay object. FeedbackDisplay macros allow access to statistics calculated on a collection of responses without requiring the user to write script and set attributes. These macros provide shortcuts for displaying statistics within a SlideText sub-object on a FeedbackDisplay object only. They are not available from script, nor can they be used in any other type of object presenting text.

Macro	Description
@ACC.MEAN	Mean accuracy for a collection of responses.
@ACC.N	Number of observations in the collection of accuracy values.
@ACC	Accuracy for an individual (i.e., the current) response.
@RT.MEAN	Mean reaction time for a collection of responses.
@RT.MIN	Minimum reaction time for a collection of responses.
@RT.MAX	Maximum reaction time for a collection of responses.
@RT.STDDEVP	Calculates standard deviation based on the population.
@RT.STDDEVS	Calculates standard deviation based on a sample.
@RT.N	Number of observations in the collection of reaction times values.
@RT	Reaction time for an individual (i.e., the current) response.
@CorrectRT.MEAN	Mean reaction time for the collection of responses for which ACC = 1 (i.e., correct responses).
@CorrectRT.MIN	Minimum reaction time for the collection of responses for which ACC = 1 (i.e., correct responses).
@CorrectRT.MAX	Maximum reaction time for the collection of responses for which ACC = 1 (i.e., correct responses).
@CorrectRT.STDDEVP	Population standard deviation of the reaction times for the collection of responses for which ACC = 1 (i.e., correct responses).
@CorrectRT.STDDEVS	Sample standard deviation of the reaction times for the collection of responses for which ACC = 1 (i.e., correct responses).
@CorrectRT.N	Number of observations in the collection of reaction times for correct responses.
@IncorrectRT.MEAN	Mean reaction time for the collection of responses for which ACC = 0 (i.e., incorrect responses).
@IncorrectRT.MIN	Minimum reaction time for the collection of responses for which ACC = 0 (i.e., incorrect responses).
@IncorrectRT.MAX	Maximum reaction time for the collection of responses for which ACC = 0 (i.e., incorrect responses).
@IncorrectRT.STDDEVP	Population standard deviation of the reaction times for the collection of responses for which ACC = 0 (i.e., incorrect responses).
@IncorrectRT.STDDEVS	Sample standard deviation of the reaction times for the collection of responses for which ACC = 0 (i.e., incorrect responses).
@ IncorrectRT.N	Number of observations in the collection of reaction times for incorrect responses.



1.3.9 InLine Object



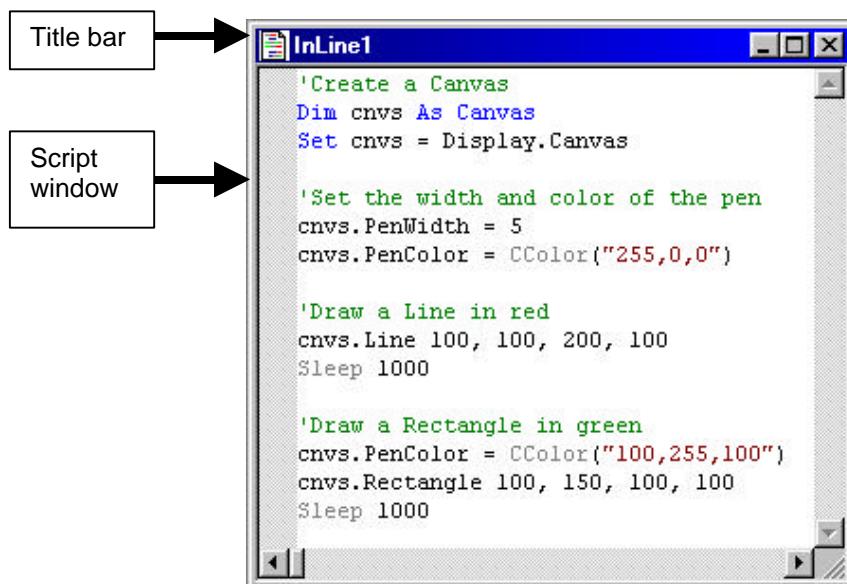
The InLine object as it appears in the Toolbox.

1.3.9.1 Overview

The InLine object is used to insert user-written E-Basic script into an E-Prime experiment. The InLine object is useful in situations in which the standard objects do not provide the flexibility required by certain paradigms, and affords the user very detailed control within the program execution. The E-Basic language is completely documented in the form of on-line Help. This Help may be accessed via the Start menu by selecting E-Basic Help from the E-Prime menu, or by using the Help menu within E-Studio. Refer to Chapter 4-Using E-Basic in the User's Guide for detailed examples concerning the use of E-Basic and writing script.

1.3.9.2 Interface

The InLine title bar appears at the top of the object window when opened in the Workspace. The title bar displays the InLine object icon, as well as the name of the InLine object. The script window appears below the title bar, and displays any user-written script.



User-written script is entered using E-Basic, the complete scripting language underlying E-Prime. Script entered into an InLine object is inserted “as-is” into the experiment script at run-time. The location of the script in the generated experiment specification file (*.ES file) is relative to the location of the InLine object within the experiment Procedures.

Script may be entered in the window by placing the cursor in the window and typing directly in the window. Alternatively, script may be entered via the Script field in the Properties window. The latter method is not recommended, as it allows the user to view only a small portion of the script at one time. E-Basic provides syntax highlighting to aid in the readability of script.



Color Defaults	Type	Description
Green	Comments	Notes entered into program code. Set off from the code by a single quotation mark (').
Blue	Keywords	Any word or symbol recognized by E-Basic as part of the language.
Black	Statements	General script.
Burgundy	Strings	A data type capable of holding a number of characters.
Gray	API functions	Application Program Interface functions (i.e., functions specific to E-Prime).
Orange	Constants	User-defined values defined as constants.

1.3.9.3 Property Pages

The InLine object is not a “runnable” object. Rather, the script contained within the object is inserted into the experiment script, as is. Therefore, only the Notes tab exists on the Property pages for the InLine object.

1.3.9.4 Properties Window

The Properties of the InLine object may be set using the fields in the Properties window.

Property	Possible Values	Description
(Name)	String values	Assigns a user-entered string as the name of the InLine object.
(About)	-----	Displays the About InLine dialog.
(Property Pages)	-----	Opens the InLine object window.
Code	E-Basic Script	User written E-Basic code which is inserted directly into the program script.
Notes	String values	User-entered text useful for storing information or comments related to an object.
Tag	String values, Attribute references.	User-defined string to be logged with the object.

1.3.10 SoundOut Object



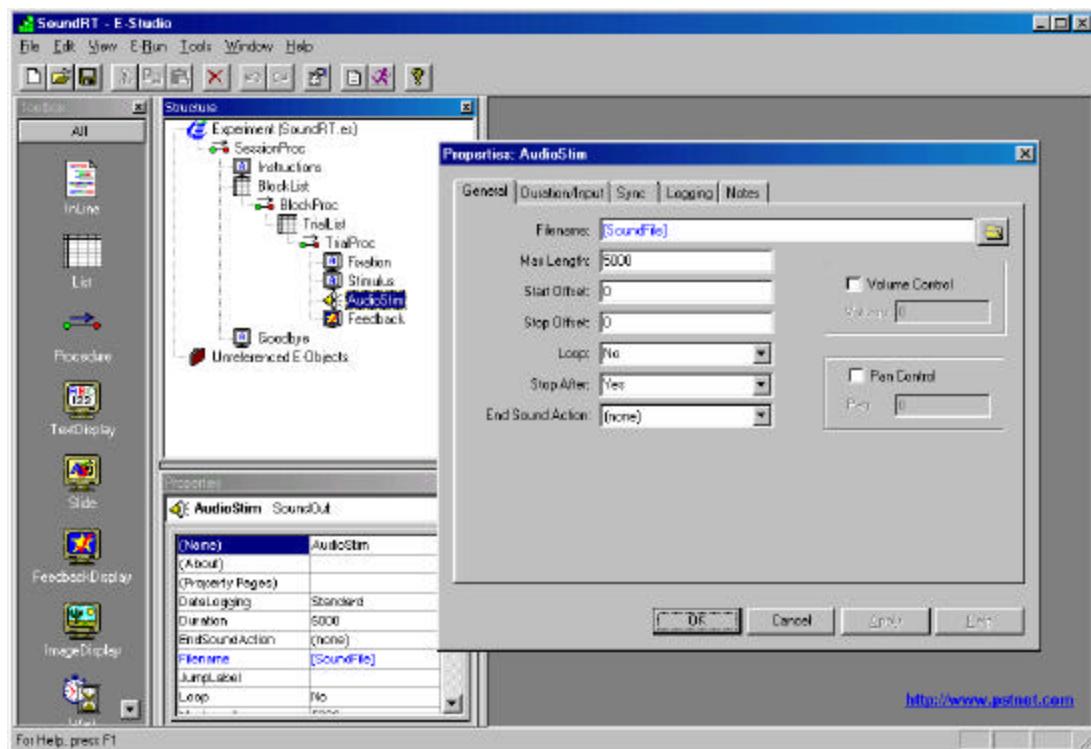
The SoundOut object as it appears in the Toolbox.

1.3.10.1 Overview

The SoundOut object is used to present pre-recorded digital audio sounds recorded in WAV file format to the subject. The SoundOut object maintains and manages a buffer on a specific sound device. The format of the WAV files must match the format properties of the Sound device.

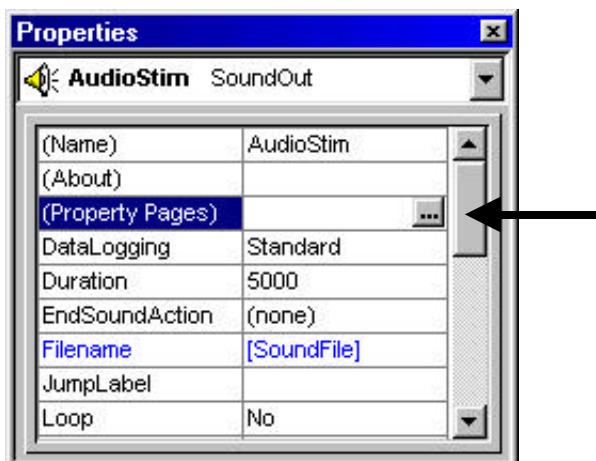
1.3.10.2 Interface

A new SoundOut object may be opened by clicking the SoundOut object icon in the Toolbox, and dragging the object to the Workspace or to a procedural timeline in the Structure view. The SoundOut object does not have an interface through which properties are set. Dragging the SoundOut object icon to the Workspace, or double clicking a SoundOut object in the Structure view will open the object’s Property pages.



1.3.10.3 Property Pages

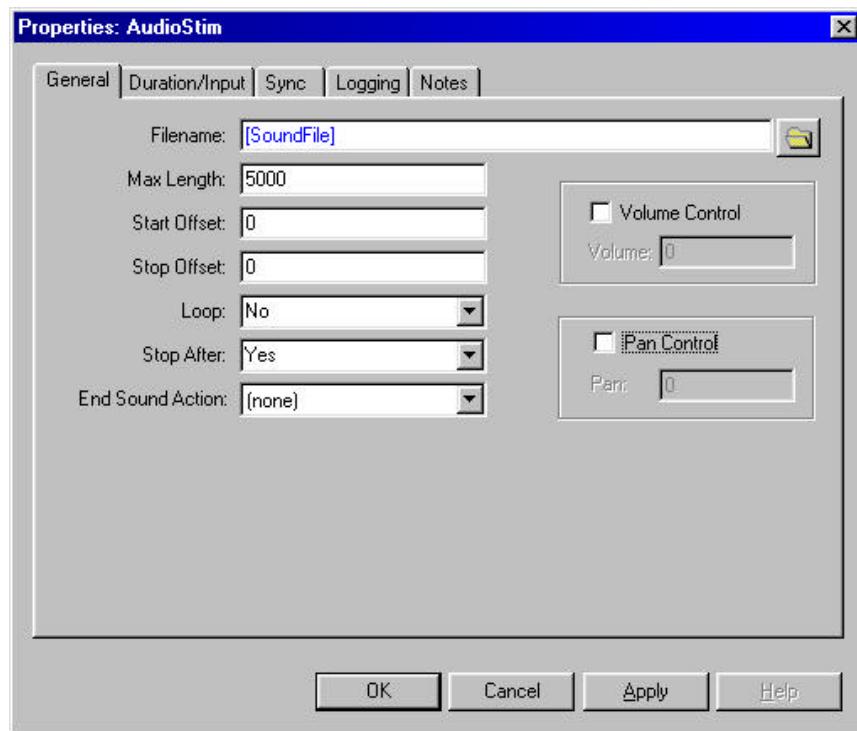
The properties for the SoundOut object may be set using the Property pages. The Property pages may be displayed by double-clicking the SoundOut object in the Structure view, or by clicking the ellipses (...) in the Property Pages field in the Properties window.





General Tab

The General tab permits the setting of properties related to the audio file presented by the SoundOut object.



Property	Possible Values	Description
Filename	Strings, Attribute references	Determines the audio file to be loaded by the SoundOut object.
MaxLength	Integer values	Determines the maximum audio buffer size.
StartOffset	Integer values, Attribute references	Determines the offset (in bytes) from which audio playback will begin.
StopOffset	Integer values, Attribute references	Determines the offset (in bytes) at which audio playback will terminate.
Loop	Yes, No	Sets playback to loop continuously between the StartOffset and StopOffset property settings.
StopAfter	Yes, No	Determines whether the playback will continue after the termination of the SoundOut object's Run method.
End Sound Action	(none), Terminate, Jump	Determines the action to be taken upon termination of the playback.
Volume Control	Yes, No	Enables or disables control of the volume level for playback.
Volume	-10000 to 0	Sets the volume level for playback.
Pan Control	Yes, No	Enables or disables panning control for playback.
Pan	-10000 to 10000	Sets the panning level (i.e., balance) for playback.

1.3.10.4 Properties Window

The Properties of the SoundOut object may be set using the fields in the Properties window. Properties to be set include information concerning the features of the SoundOut (e.g., PanControl, Volume), as well as the duration of the SoundOut, whether input is allowed, and the termination conditions for the playback of the SoundOut.



SoundOut Properties Continued...

Property	Possible Values	Description
(Name)	String values	Assigns a user-entered string as the name of the SoundOut object.
(About)	-----	Displays the About SoundOut dialog.
(Property Pages)	-----	Opens the Property pages for the SoundOut object.
DataLogging	None, Standard, Response Only, Time Audit Only, Custom	Determines the category of variables to be logged by the object.
Duration	-1 (forever), Numeric values (default in ms), Attribute references.	Determines the duration of the object in milliseconds.
EndSoundAction	(none), Terminate, Jump	Determines the action to be taken upon termination of the audio playback.
Filename	String values (e.g., <i>Filename.WAV</i>), Attribute references.	Designates the name of the audio file to play.
JumpLabel	String values, Attribute references.	Identifies a Label object within the current Procedure to which program execution jumps when input is received on a mask that has an End Action of "Jump".
Loop	Yes, No, Attribute references.	Determines whether the audio playback continues in a loop, or stops at the end of the data playback.
MaxLength	Positive numeric values	Defines the maximum size of the buffer in milliseconds. Max length does not accept attribute references, and is read-only at run-time.
Notes	String values	User-entered text useful for storing information or comments related to an object.
OffsetSync	(none), Vertical blank	Sets the object or event with which the offset of the object is synchronized.
OnsetSync	(none), Vertical blank	Sets the object or event with which the onset of the object is synchronized.
Pan	Numeric values, Attribute references.	Specifies the pan level in 100th decibels. Functionality is dependent on the PanControl property setting.
PanControl	Yes, No	Enables or disables panning control.
PreRelease	Numeric values (default in msec), Attribute references.	Amount of time released during the processing of the current object to allow for setup of the next object.
StartOffset	Specific string values (e.g., "0"), Attribute references.	Determines the offset (in bytes) from which audio playback will begin.
StopAfter	Yes, No, Attribute references.	Determines whether the playback will continue after the termination of the SoundOut object's Run method.
StopOffset	Specific string values (e.g., "0"), Attribute references.	Determines the offset (in bytes) from which audio playback will terminate.
Tag	String values, Attribute references.	User-defined string that is useful for associating information with an object.
TimingMode	Event, Cumulative, Custom	Sets the timing method to be used by the object. Cumulative timing is used to absorb the processing time between events, and to maintain a certain interval duration between events or between trials.
Volume	Numeric values, Attribute references.	Sets the volume of the SoundOut buffer. Currently specifies 100th decibels. Functionality is dependent upon the VolumeControl property setting.
VolumeControl	Yes, No	Enables or disables volume control by the SoundOut object.



1.3.11 Wait Object



The Wait object as it appears in the Toolbox.

1.3.11.1 Overview

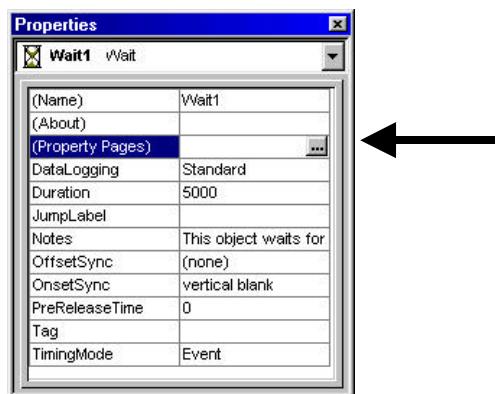
The Wait object is used to delay program execution for a specified amount of time. The Wait object may also be used to collect a response without requiring the overhead of an object presenting a stimulus (e.g., TextDisplay, SoundOut, etc.).

1.3.11.2 Interface

The Wait object does not have an interface through which properties are set. Dragging the Wait object icon to the Workspace, or double clicking a Wait object in the Structure view will open the object's Property pages.

1.3.11.3 Property Pages

The properties for the Wait object may be set using the Property pages. The Property pages may be displayed by double-clicking the Wait object in the Structure view, or by clicking the ellipses (...) in the Property Pages field in the Properties window. There are no Property pages specific to the Wait object. Refer to section 1.3.1 for a discussion of standard property pages.



1.3.11.4 Properties Window

The properties specific to the Wait object are included in the table below.

Property	Possible Values	Description
(Name)	String values	Assigns a user-entered string as the name of the ImageDisplay.
(About)	-----	Displays the About ImageDisplay dialog.
(Property pages)	-----	Opens the Property pages for the ImageDisplay.
DataLogging	None, Standard, Response Only, Time Audit Only, Custom	Determines the category of variables logged for the object.
Duration	-1 (forever), Numeric values (default in msec), Attribute references.	Determines the duration of the object in milliseconds.



Wait Properties Continued...

JumpLabel	String values	Identifies a Label object within the current Procedure to which program execution jumps when input is received on a mask that has an End Action of "Jump".
Notes	String values	User-entered text useful for storing information or comments related to an object.
OffsetSync	(none), Vertical blank	Sets the object or event with which the offset of the object is synchronized.
OnsetSync	(none), Vertical blank	Sets the object or event with which the onset of the object is synchronized.
PreRelease	Integer values	Amount of time released during the processing of the current object to allow for setup of the next object.
Tag	String values, Attribute references.	User-defined string that is useful for associating information with an object.
TimingMode	Event, Cumulative, Custom	Sets the timing method to be used by the object. Cumulative timing is used to absorb the processing time between events, and to maintain a certain interval duration between events or between trials.

1.3.12 Label Object



The Label object as it appears in the Toolbox.

1.3.12.1 Overview

The Label object is used to mark a position in a procedural timeline to which processing moves in response to the "Jump" End Action option for an input object. Jumping is an action option in response to the termination of an object's input mask. Jumping is possible only within a single Procedure. It is not possible to jump out of one Procedure to a Label in another Procedure.

1.3.12.2 Interface

There is no interface specific to the Label object. Dragging the Label object icon to the Workspace, or double clicking a Label object in the Structure view will open the object's Property pages.

1.3.12.3 Property Pages

The Property pages may be displayed by double-clicking the Label object in the Structure view, or by clicking the ellipses (...) on the Property Pages field in the Properties window. There are no Property pages specific to the Label object. Refer to section 1.3.1 for a description of standard property pages.

1.3.12.4 Properties Window

Property	Possible Values	Description
(Name)	String values	Assigns a user-entered string as the name of the Label.
(About)	-----	Displays the About Label dialog.
(Property Pages)	-----	Displays the Property pages for the object.
Notes	String values	User-entered text useful for storing information or comments related to an object.
Tag	String values, Attribute references.	User-defined string useful for associating information with an object.



1.3.13 PackageCall Object



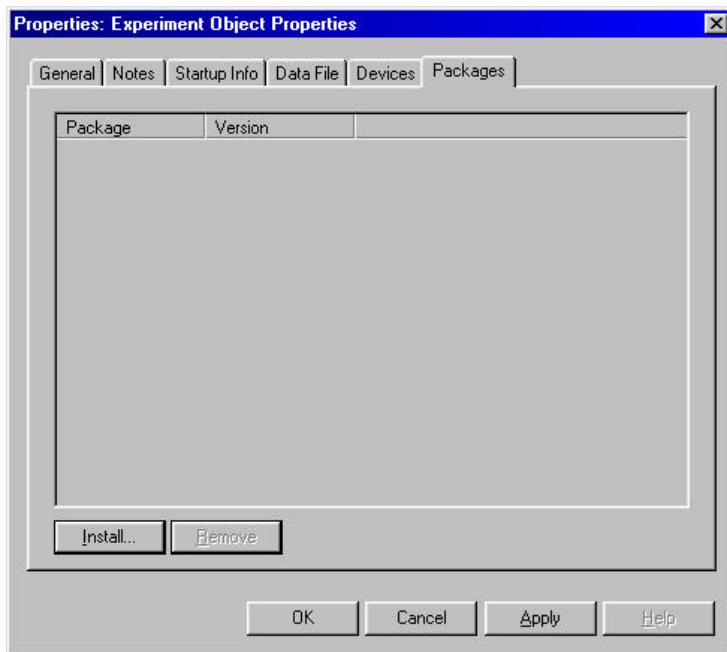
The Package Call object as it appears in the Toolbox.

1.3.13.1 Overview

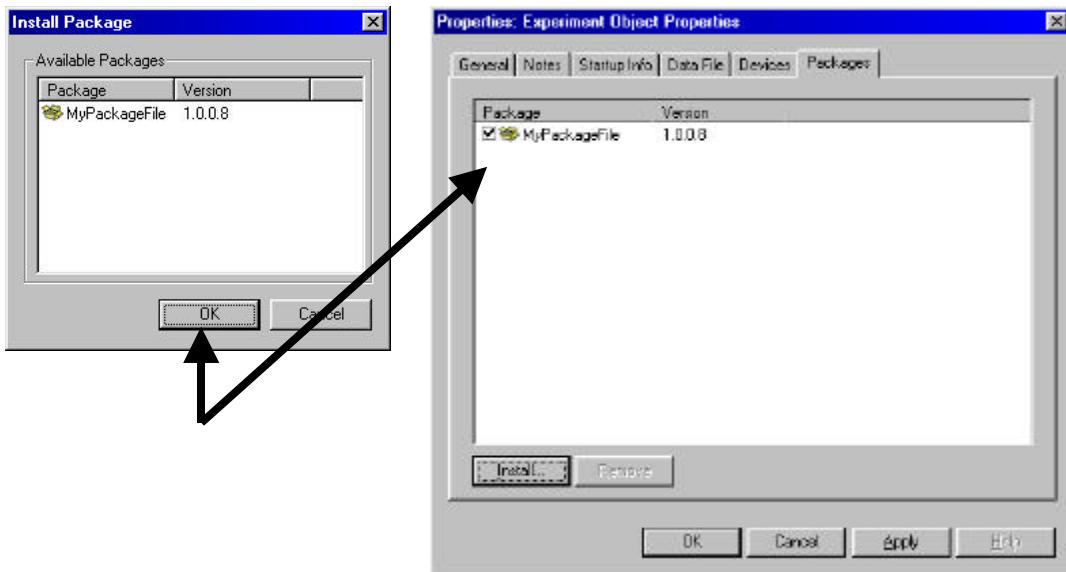
The PackageCall object permits a block of E-Basic script to be loaded into the Experiment Specification file from a text file. The PackageCall object is most useful when a segment of script is used by multiple experiments. The use of package files reduces the copying, pasting, and maintenance of inserting script that is reused often. PackageCall objects refer to package files (*.EPK, text files) containing E-Basic script that adheres to a required format. Package files define one or more code routines that may be accessed through a PackageCall object once the appropriate package file is installed in the program.

Installing Package Files

To install a Package file in a program, it must be added to the experiment via the Packages tab in the Experiment object Property pages. Double click the Experiment object in the Structure view to display the Property pages, and select the Packages tab.



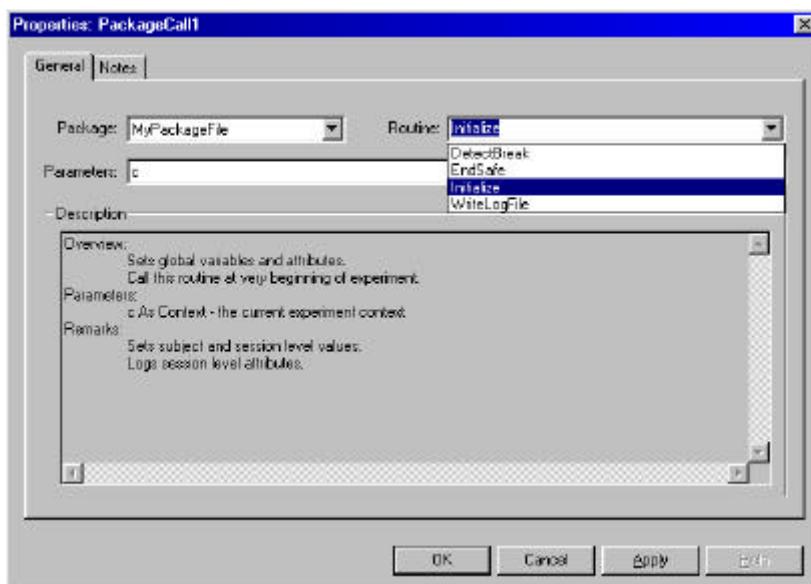
Each package file must exist in a unique subdirectory within the Packages folder portion of the E-Prime installation (default location is C:\Program Files\PST\E-Prime\Program\Packages). When the Install button is clicked, all packages available for installation will be displayed. Select the appropriate file in the Package column and click OK. Once installed, the package file's icon will be displayed on the Experiment object's Packages tab.



The checkbox next to the package file's icon allows a specific package file to be enabled or disabled without uninstalling or reinstalling. When a package file is disabled, no script will be inserted from the file during the compile process.

Using Package Files

After the appropriate package file is installed in the Experiment object, the code routines contained within the package file may be accessed using a PackageCall object. The PackageCall object is equivalent to inserting a segment of user-written script into a Procedure via an InLine object. The placement of the PackageCall in the experiment's Procedures determines where the script called by the PackageCall is inserted in the complete experiment script when generated.





1.3.13.2 Interface

There is no interface specific to the PackageCall object. Dragging the PackageCall object icon to the Workspace, or double clicking a PackageCall object in the Structure view will open the object's Property pages for editing.

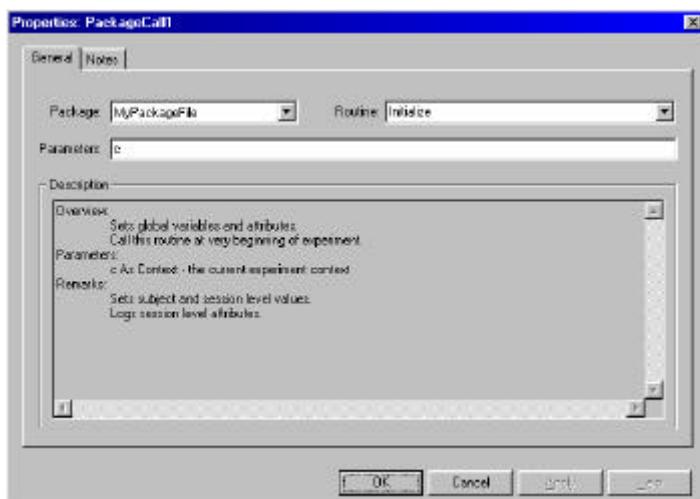
1.3.13.3 Property Pages

The properties for the PackageCall object may be set using the Property pages. The Property pages may be displayed by double-clicking the PackageCall object in the Structure view, or by clicking the ellipses (...) in the Property Pages field in the Properties window.



General Tab

The General tab permits the identification of the Package file and routine to be called, as well as the parameters required by that routine.



Property	Possible Values	Description
Package	Installed EPK files, selected from dropdown	Name of the Package file to be called.
Routine	Subroutines defined in the package file, selected from dropdown	Name of the routine to be called.
Parameters	Package-dependent	Parameters passed to the routine.
Description	Read only	User defined string that is useful for associating information with an object.



1.3.13.4 Properties Window

Property	Possible Values	Description
(Name)	String values	Assigns a user-entered string as the name of the object.
(About)	-----	Displays the About Label dialog (currently not implemented).
(Property Pages)	-----	Displays the Property pages for the object.
Notes	String values	User-entered text useful for storing information or comments related to an object.
Package	Installed EPK files, selected from dropdown	Name of the Package file to be called.
Parameters	Package-dependent	Parameters passed to the routine.
Routine	Suboutines defined in the package file, selected from dropdown	Name of the routine to be called.
Tag	String values, Attribute references.	User-defined string that is useful for associating information with an object.

1.3.14 Scripting Objects

Scripting objects differ from E-Objects in that there is no graphical user interface or Property pages with which to set or determine properties. All properties and methods for scripting objects are accessible only through user-written script in an InLine object or in the Script window. Some commonly used scripting objects are described below. Refer to the E-Basic on-line Help for a complete listing of objects, and properties and methods associated with each object.

1.3.14.1 Canvas Object

The Canvas abstracts the programming interface to a particular canvas/page/surface. As such, there is no interface for the Canvas object itself, and all access occurs through script. Canvas objects support all of the drawing/graphics calls that are available in the system.

1.3.14.2 Summation Object

Summation objects are used to collect a series of observations. From this collection, various summary measures may be determined, such as the minimum or maximum value in the collection, the number of observations, and various statistical measures, such as the mean, standard deviation, or variance of the observations within the collection. For example, in order to determine overall accuracy in a block of trials, a Summation object may be used to keep track of the individual observations, and to calculate the desired measure from the total collection.

1.3.14.3 Debug Object

The Debug object encapsulates a set of useful debugging mechanisms. The Debug.Print command sends a string to the Debug tab in the Output window at run-time, which is helpful when verifying sampling sequences or timing presentation of an object. The Debug commands may be used when developing or testing a new program. Examples of the Debug.Print command can be found in the Using E-Basic chapter of the User's Guide and also within the Advanced Tutorials of the Getting Started Guide.



This page intentionally left blank.



Chapter 2: E-Basic

E-Basic is the complete scripting language underlying E-Prime. E-Basic is almost identical to Visual Basic for Applications, with additional commands inserted to accommodate the needs of empirical research. For detailed information concerning E-Basic commands, refer to the on-line Help. The E-Basic on-line Help may be accessed via the E-Prime menu.

2.1 Listing of E-Basic Commands

The following is a list of E-Basic commands grouped by type. These commands may be used in an InLine object, and are used by E-Studio to convert object specifications into script.

2.1.1 *Audio Control*

SoundBuffer.Continue (method)	SoundBuffer.Play (method)
SoundDevice.CreateBuffer (method)	SoundOut.Play (method)
SoundBuffer.Load (method)	SoundOut.Run (method)
SoundDevice.Open (method)	Slide.Stop (method)
SoundBuffer.Pause (method)	SlideSoundOut.Stop (method)
Slide.Play (method)	SlideState.Stop (method)
SlideSoundOut.Play (method)	SoundBuffer.Stop (method)
SlideState.Play (method)	SoundOut.Stop (method)

2.1.2 *Compiler Control*

#Const (directive)	Inline (statement)
#If... Then... #Else (directive)	Option Default (statement)
() (keyword)	Option Explicit (statement)
_ (keyword)	

2.1.3 *Data Logging*

Summation.AddObservation (method)	DataFile.Open (method)
DataFile.Close (method)	Summation.RemoveObservation (method)
DataFile.Convert (method)	FeedbackDisplay.Reset (method)
DataFile.Flush (method)	Summation.StdDevP (method)
Summation.Max (method)	Summation.StdDevS (method)
Summation.Mean (method)	Summation.Total (method)
Summation.Min (method)	Summation.VarP (method)
Summation.N (method)	Summation.VarS (method)



2.1.4 Data Type Conversion

Asc (function)	CSng (function)
AscB (function)	CStr (function)
AscW (function)	CString (function)
CBool (function)	CVar (function)
CColor (function)	CVDate (function)
CColorPalette (function)	CVErr (function)
CCur (function)	Fix (function)
CDate (function)	Hex (function)
CDbl (function)	Hex\$ (function)
Chr (function)	Int (function)
Chr\$ (function)	IsDate (function)
ChrB (function)	IsNumeric (function)
ChrB\$ (function)	Oct (function)
ChrW (function)	Oct\$ (function)
ChrWS\$ (function)	RGB (function)
CInt (function)	Str (function)
CLng (function)	Str\$ (function)
CLogical (function)	Val (function)

2.1.5 Data Type Conversion: Casting

CCanvas (function)	CPortInputDevice (function)
CCellSpec (function)	CPortresponseData (function)
CContext (function)	CRandomOrder (function)
CCycles (function)	CRandomReplaceOrder (function)
CDataFile (function)	CResponseData (function)
CDeletion (function)	CResponseDataCollection (function)
CDevice (function)	CRteCollection (function)
CDeviceManager (function)	CSamples (function)
CDisplayDevice (function)	CSequentialOrder (function)
CDisplayEchoClient (function)	CSerialDevice (function)
CDisplayEchoClientCollection (function)	CSlide (function)
CEchoClient (function)	CSlideImage (function)
CExplicitOrder (function)	CSlideSoundOut (function)
CExplicitOrderByName (function)	CSlideState (function)
CFactor (function)	CSlideStateCollection (function)
CFactorSpace (function)	CSlideStim (function)
CFactorTable (function)	CSlideStimCollection (function)
CFeedbackDisplay (function)	CSlideText (function)
CImageDisplay (function)	CSlideVisualStim (function)
CInputDevice (function)	CSoundBuffer (function)
CInputHistoryManager (function)	CSoundBufferCollection (function)
CInputMask (function)	CSoundDevice (function)
CInputMaskManager (function)	CSoundOut (function)
CKeyboardInputDevice (function)	CSRBoxEchoClient (function)
CKeyboardresponseData (function)	CSRBoxEchoClientCollection (function)
CLatinSquares (function)	CSRBoxInputDevice (function)
CMouseInputDevice (function)	CSRBoxResponseData (function)
CMouseresponseData (function)	CStimDisplay (function)
CNoDeletion (function)	CSummation (function)
COffsetOrder (function)	CTextDisplay (function)
COrder (function)	CTimed (function)
CPermutationOrder (function)	CTrigger (function)
CPickOne (function)	



2.1.6 Date and Time

Date (function)	Month (function)
Date (statement)	Now (function)
Date\$ (function)	Second (function)
Date\$ (statement)	Time (function)
DateAdd (function)	Time (statement)
DateDiff (function)	Time\$ (function)
DatePart (function)	Time\$ (statement)
DateSerial (function)	Timer (function)
DateValue (function)	TimeSerial (function)
Day (function)	TimeValue (function)
Hour (function)	Weekday (function)
Minute (function)	Year (function)

2.1.7 Display: Dialogs

AnswerBox (function)	MsgBox (function)
AskBox (function)	MsgBox (statement)
AskBox\$ (function)	OpenFilename\$ (function)
AskPassword (function)	PopupMenu (function)
AskPassword\$ (function)	SaveFilename\$ (function)
InputBox (function)	SelectBox (function)
InputBox\$ (function)	

2.1.8 Display: General Control

EchoClient.AddEchoData (method)	DisplayDeviceEchoClient.Draw (method)
DisplayDevice.CalcActualRefreshRate (method)	Slide.Draw (method)
DisplayDevice.CalculateRefreshRate (method)	SlideState.Draw (method)
DisplayDevice.Canvas (method)	SlideVisualStim.Draw (method)
Slide.Clear (method)	StimDisplay.Draw (method)
SlideState.Clear (method)	GetDefaultDisplayDevice (function)
SlideVisualStim.Clear (method)	SlideState.HitTest (method)
StimDisplay.Clear (method)	ImageDisplay.Load (method)
CPaletteIndex (function)	SlideImage.Load (method)
DisplayDevice.CreateCanvas (method)	DisplayDevice.Open (method)
DisplayDeviceEchoClientCollection.CreateEchoClient (method)	DisplayDevice.Palette (method)
SRBoxDeviceEchoClientCollection.CreateEchoClient (method)	DisplayDevice.RestoreVideoBuffers (method)
EchoClient.Detach (method)	DisplayDevice.WaitForVerticalBlank (method)

2.1.9 Display: Graphics

Canvas.Arc (method)	Canvas.MoveTo (method)
Canvas.CalculateTextSize (method)	Canvas.Oval (method)
Canvas.Chord (method)	Canvas.Pie (method)
Canvas.Circle (method)	PointInRect (function)
Canvas.Clear (method)	Canvas.Polygon (method)
Canvas.Copy (method)	Canvas.Rectangle (method)
Palette.GetEntries (method)	Canvas.RoundedRectangle (method)
Canvas.GetPixel (method)	Canvas.SaveImage (method)
Canvas.Line (method)	Palette.SetEntries (method)
Canvas.LineTo (method)	Canvas.SetPixel (method)
Canvas.LoadImage (method)	Canvas.Text (method)



2.1.10 Error Handling

Err.Clear (method)	Err.HelpFile (property)
Err.Description (property)	Err.LastDLLError (property)
Erl (function)	Err.Number (property)
Error (statement)	On Error (statement)
Error (function)	Err.Raise (method)
Error\$ (function)	Resume (statement)
Err.HelpContext (property)	Err.Source (property)

2.1.11 Experiment Control: Ports

SerialDevice.FlushInputBuffer (method)	ReadPort (function)
SerialDevice.FlushOutputBuffer (method)	SerialDevice.ReadString (method)
SerialDevice.Open (method)	IOPort.Write (method)
SerialDevice.Peek (method)	SerialDevice.WriteByte (method)
IOPort.Read (method)	SerialDevice.WriteLine (method)
SerialDevice.ReadBytes (method)	SerialDevice.WriteLong (method)
SerialDevice.ReadInteger (method)	WritePort (statement)
SerialDevice.ReadLong (method)	SerialDevice.WriteString (method)

2.1.12 Experiment Control: Stimulus

CellSpec.Add (method)	Order.IsEmpty (method)
Context.Add (method)	Factor.Load (method)
FactorSpace.AddAttrib (method)	Context.Log (method)
FactorTable.AddFactor (method)	Context.LogHeader (method)
Factor.AddLevel (method)	Order.PeekAhead (method)
Order.Advance (method)	FactorSpace.PeekAttrib (method)
Context.AttribExists (method)	Context.PopFrame (method)
Deletion.Count (method)	Context.PushNewFrame (method)
Order.Count (method)	FactorSpace.Reset (method)
Context.GetAttrib (method)	RteRunnableObject.Run (method)
Factor.GetAttrib (method)	Context.SetAttrib (method)
FactorSpace.GetAttrib (method)	Factor.SetAttrib (method)
FactorSpace.GetAttribLogFlag (method)	Context.SetAttribAtSource (method)
Context.GetAttribNames (method)	FactorSpace.SetAttribLogFlag (method)
Context.GetCellSpec (method)	Context.SetLogLevelName (method)
Order.GetCurrent (method)	Factor.SetNested (method)
FactorSpace.GetCurrentAttrib (method)	FactorSpace.SetNested (method)
Order.GetFactorSpace (method)	Factor.SetProc (method)
Context.GetLevel (method)	FactorSpace.SetProc (method)
Context.GetLevelName (method)	Factor.SetWeight (method)
Context.GetLogLevelName (method)	FactorSpace.Size (method)
Order.GetNext (method)	FactorTable.Size (method)
FactorSpace.GetNextAttrib (method)	FactorSpace.Terminate (method)
FactorSpace.GetPrevAttrib (method)	LatinSquares.TotalRunsNeeded (method)
Order.GetPrevious (method)	LatinSquares.TotalSamplesNeeded (method)
FactorSpace.GetProc (method)	Context.Update (method)

2.1.13 Experiment Control: Timing

GetNextTargetOnsetTime (function)	Clock.ReadMillsec (method)
GetOSThreadPriority (function)	Clock.ResetScale (method)
Clock.Read (method)	SetNextTargetOnsetTime (statement)
Clock.Read1000thMillsec (method)	SetOSThreadPriority (statement)
Clock.Read100thMillsec (method)	SetPcodeSleepDuration (statement)
Clock.Read10thMillsec (method)	SetPcodeSleepFrequency (statement)
Clock.ReadMicrosec (method)	



2.1.14 File Input/Output

Close (statement)	Lock (statement)
Eof (function)	Lof (function)
FreeFile (function)	Open (statement)
Get (statement)	Print # (statement)
Input (function)	Put (statement)
Input# (statement)	Reset (statement)
Input\$ (function)	Seek (function)
InputB (function)	Seek (statement)
InputB\$ (function)	Unlock (statement)
Line Input # (statement)	Width# (statement)
Loc (function)	Write# (statement)

2.1.15 Input Device Control

InputDevice.CreateInputMask (method)	MouseResponseData.IsButton8 (method)
SRBoxDevice.Flush (method)	MouseDevice.IsCursorVisible (method)
MouseDevice.GetCursorLimits (method)	SRBoxDevice.KeyCount (method)
MouseDevice.GetCursorPos (method)	SRBoxDevice.Lamps (method)
InputDevice.InsertResponse (method)	KeyboardDevice.Open (method)
MouseResponseData.IsButton1 (method)	MouseDevice.Open (method)
MouseResponseData.IsButton2 (method)	PortDevice.Open (method)
MouseResponseData.IsButton3 (method)	SRBoxDevice.Open (method)
MouseResponseData.IsButton4 (method)	MouseDevice.SetCursorLimits (method)
MouseResponseData.IsButton5 (method)	MouseDevice.SetCursorPos (method)
MouseResponseData.IsButton6 (method)	MouseDevice.ShowCursor (method)
MouseResponseData.IsButton7 (method)	

2.1.16 Math

Abs (function)	Sgn (function)
Atn (function)	Sin (function)
Cos (function)	Sqr (function)
Exp (function)	Tan (function)
Log (function)	

2.1.17 Math: Financial

DDB (function)	Pmt (function)
FV (function)	PPmt (function)
IPmt (function)	PV (function)
IRR (function)	Rate (function)
MIRR (function)	SLN (function)
NPer (function)	SYD (function)
NPV (function)	

2.1.18 Math: Randomization

PRNG.GetNext (method)	Randomize (statement)
PRNG.GetPrev (method)	RandomizeArray (function)
PRNG.GetSeed (method)	PRNG.Reset (method)
PRNG.Random (method)	Rnd (function)
Random (function)	PRNG.SetSeed (method)
Random (function)	SetSeed (statement)
Randomize (function)	



2.1.19 Object Control

CreateObject (function)	Is (operator)
GetObject (function)	Nothing (keyword)

2.1.20 Operators

- (operator)	> (operator)
* (operator)	>= (operator)
/ (operator)	And (operator)
\ (operator)	Eqv (operator)
^ (operator)	Imp (operator)
+ (operator)	Mod (operator)
< (operator)	Not (operator)
<= (operator)	Or (operator)
<> (operator)	Xor (operator)
= (operator)	

2.1.21 Program Control

RteCollection.Add (method)	If...Then...Else (statement)
Debug.Assert (method)	IIf (function)
Call (statement)	InputMask.IsPending (method)
Choose (function)	InputMaskManager.IsPending (method)
Device.Close (method)	DeviceManager.IsSuspended (method)
Declare (statement)	Main (statement)
Do...Loop (statement)	Device.Open (method)
DoEvents (function)	Debug.Print (method)
DoEvents (statement)	RteCollection.Remove (method)
End (statement)	RteCollection.RemoveAll (method)
Exit Do (statement)	InputMaskManager.Reset (method)
Exit For (statement)	Device.Resume (method)
Exit Function (statement)	DeviceManager.Resume (method)
Exit Sub (statement)	Return (statement)
For Each...Next (statement)	Select...Case (statement)
For...Next (statement)	SetUserBreakState (statement)
Function...End (statement)	Sleep (statement)
DeviceManager.GetDevice (method)	Sleep (statement)
DeviceManager.GetDeviceCount (method)	Sub...End Sub (statement)
Rte.GetObject (method)	Device.Suspend (method)
Rte.GetObjectCount (method)	DeviceManager.Suspend (method)
Device.GetState (method)	Switch (statement)
GetUserBreakState (function)	InputMask.Terminate (method)
GoSub (statement)	InputMaskManager.Terminate (method)
Goto (statement)	While...Wend (statement)



2.1.22 String Manipulation

& (operator)	MidB (function)
Format (function)	MidB (statement)
Format\$ (function)	MidB\$ (function)
InStr (function)	MidB\$ (statement)
InStrB (function)	Option Compare (statement)
Item\$ (function)	Option CStrings (statement)
ItemCount (function)	Right (function)
LCASE (function)	Right\$ (function)
LCASE\$ (function)	RightB (function)
Left (function)	RightB\$ (functions)
Left\$ (function)	RSet (statement)
LeftB (function)	RTrim (function)
LeftB\$ (function)	RTrim\$ (function)
Len (function)	Space (function)
LenB (function)	Space\$ (function)
Like (function)	StrComp (function)
Line\$ (function)	StrConv (function)
LineCount (function)	String (function)
LSet (function)	String\$ (function)
LTrim (function)	Trim (function)
LTrim\$ (function)	Trim\$ (function)
Mid (function)	UCase (function)
Mid (statement)	UCase\$ (function)
Mid\$ (function)	Word\$ (function)
Mid\$ (statement)	WordCount (function)

2.1.23 Variable Control

= (statement)	Global (statement)
ArrayDims (function)	IsEmpty (function)
ArraySort (statement)	IsError (function)
Const (statement)	IsMissing (function)
DefBool (statement)	IsNull (function)
DefCur (statement)	isObject (function)
DefDate (statement)	LBound (function)
DefDbl (statement)	Let (statement)
DefInt (statement)	Option Base (statement)
DefLng (statement)	Private (statement)
DefObj (statement)	Public (statement)
DefSng (statement)	ReDim (statement)
DefStr (statement)	Set (statement)
DefVar (statement)	Type (statement)
Dim (statement)	UBound (function)
Erase (statement)	VarType (function)



**E-Prime Reference Guide
Chapter 2: E-Basic**

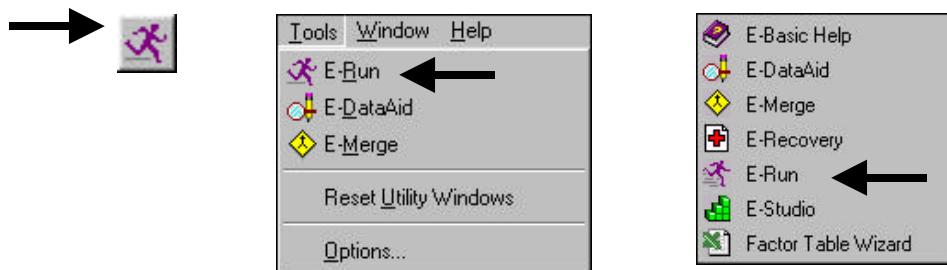
This page intentionally left blank.



Chapter 3: E-Run

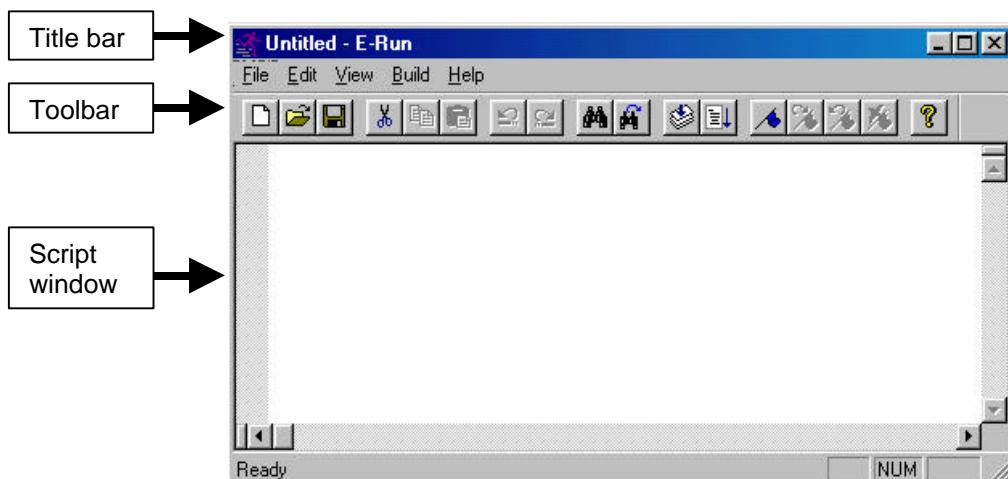
3.1 About E-Run

E-Run is the data collection application within E-Prime. When an E-Prime experiment is **generated**, E-Studio converts the experiment specification (*.ES file) from the objects specified in E-Studio to the actual E-Basic script (*.EBS file) necessary to run the program. That experiment script (*.EBS file) is then **compiled** so that the computer can interpret the instructions. E-Run is the application which processes the code in the EBS file in order to run the program and collect data. E-Run may be launched using the Run tool button on the toolbar in E-Studio, through the E-Run command in the Tools menu in E-Studio, or through the E-Prime menu from the Start menu.



3.2 About the Interface

The title bar is displayed at the top of the application screen, and displays the E-Run icon, the name of the EBS file, and the name of the application (i.e., E-Run). The menu bar is displayed below the title bar. The menu bar lists the menus available within the E-Run application, including the File, Edit, View, Build, and Help menus.





3.2.1 Menus

3.2.1.1 File Menu

Command	Shortcut	Function
<u>New</u>	Ctrl+N	Opens a new E-Run (EBS) file.
<u>Open...</u>	Ctrl+O	Opens an existing EBS file in E-run.
<u>Save</u>	Ctrl+S	Saves the E-Basic script file with the current name.
Save <u>As...</u>		Saves the E-Basic script file (EBS) with the designated name.
<u>Print...</u>	Ctrl+P	Sends the EBS file to the printer.
Print Preview		Displays how the EBS file will appear when printed.
Print Setup...		Sets the print parameters without actually sending the job to the printer.
Recent File History		Lists the most recently opened EBS files.
<u>Exit</u>		Exits the E-Run application.

3.2.1.2 Edit Menu

Command	Shortcut	Function
<u>Undo</u>	Ctrl+Z	Reverses the last operation.
<u>Redo</u>	Ctrl+Y	Repeats the last operation.
<u>Cut</u>	Ctrl+X	Removes the selected area and places it in the clipboard.
<u>Copy</u>	Ctrl+C	Copies the selected area to the clipboard.
<u>Paste</u>	Ctrl+V	Inserts the contents of the clipboard.
<u>Find...</u>	Ctrl+F	Searches for a specified string.
Find next	F3	Searches for the next instance of the specified string by either 1) matching a whole word, 2) matching a case or 3) whole expression.
<u>Replace...</u>	Ctrl+H	Searches for a specified string by either 1) matching a whole word, 2) matching a case or 3) whole expression, then replaces it with the specified replacement string.
<u>Go To...</u>	Ctrl+G	Repositions the cursor at the specified line number.

3.2.1.3 View Menu

Command	Function
<u>Toolbar</u>	Turns the display of the toolbar on or off depending on its current state. A checkmark next to the toolbar command indicates the toolbar is displayed.
<u>Status bar</u>	Turns the display of the Status bar on or off depending on its current state. A checkmark next to the Status bar command indicates the Status bar is displayed.

3.2.1.4 Build Menu

Command	Shortcut	Function
<u>Compile</u>	Ctrl+F7	Takes the generated EBS file from E-Studio and feeds it into the E-Basic compiler. The process of compiling checks the syntax of the EBS file contents to make sure that it is in an acceptable format to run.
<u>Run</u>	F7	Executes the E-Basic script.

3.2.1.5 Help Menu

Command	Function
<u>About E-Run...</u>	Displays a dialog containing the application's version information.

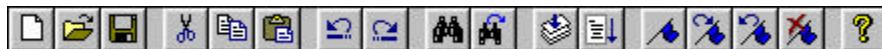


3.2.1.6 Context Menu

A context menu may be accessed by right-clicking within the Script window. The context menu includes the Cut, Copy, Paste, and Select All commands.

3.2.2 Toolbar

The toolbar is located at the top of the application display, below the menu bar. The toolbar includes shortcuts to frequently used functions in E-Run.

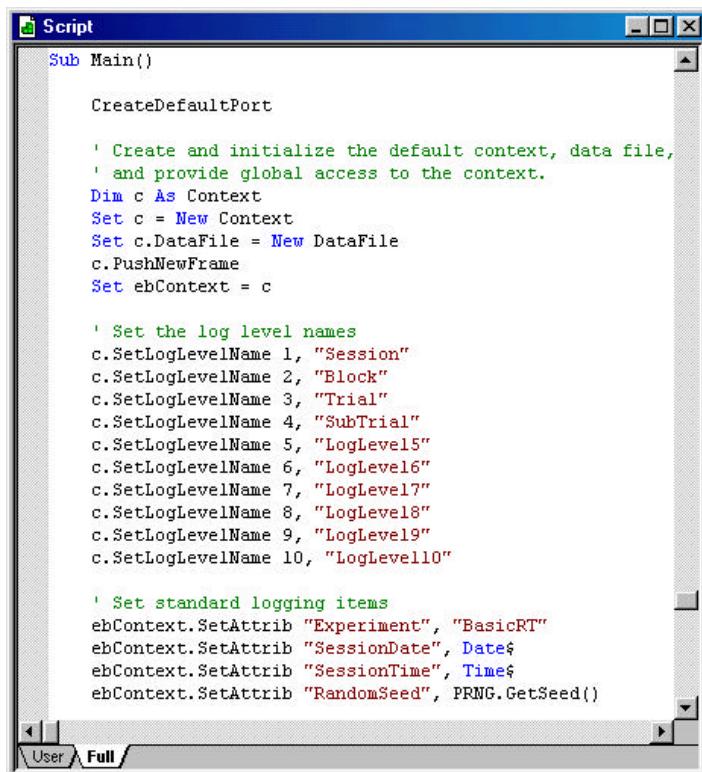


Tool Button	Description
	Opens a new E-Basic script file in E-Run.
	Displays the Open dialog to navigate to an existing EBS file.
	Saves the open EBS file.
	Cuts the selected text.
	Copies the selected text to the clipboard.
	Pastes the contents of the clipboard into the EBS file.
	Reverses the last operation.
	Repeats the last operation.
	Displays the Find dialog to search for specific text.
	Locates the next instance of the selected text.
	Compiles the E-Basic script file.
	Runs the open EBS file.
	Marks or removes marking of a line within the EBS file.
	Locates the next bookmark in the file.
	Locates the previous bookmark in the file.
	Clears all bookmarks.
	Displays the About E-Run dialog.



3.2.3 Script window

The Script window displays the E-Basic script generated by E-Studio into the EBS file. Within an EBS file, the script is color-coded for informational purposes and ease of reading.



The screenshot shows a Windows-style application window titled "Script". The main area contains E-Basic code. The code includes several comments in green, keywords in blue, statements in black, and strings in burgundy. The code is as follows:

```
Sub Main()

    CreateDefaultPort

    ' Create and initialize the default context, data file,
    ' and provide global access to the context.
    Dim c As Context
    Set c = New Context
    Set c.DataFile = New DataFile
    c.PushNewFrame
    Set ebContext = c

    ' Set the log level names
    c.SetLogLevelName 1, "Session"
    c.SetLogLevelName 2, "Block"
    c.SetLogLevelName 3, "Trial"
    c.SetLogLevelName 4, "SubTrial"
    c.SetLogLevelName 5, "LogLevel5"
    c.SetLogLevelName 6, "LogLevel6"
    c.SetLogLevelName 7, "LogLevel7"
    c.SetLogLevelName 8, "LogLevel8"
    c.SetLogLevelName 9, "LogLevel9"
    c.SetLogLevelName 10, "LogLevel10"

    ' Set standard logging items
    ebContext.SetAttrib "Experiment", "BasicRT"
    ebContext.SetAttrib "SessionDate", Date$
    ebContext.SetAttrib "SessionTime", Time$
    ebContext.SetAttrib "RandomSeed", PRNG.GetSeed()

End Sub
```

Color Defaults	Type	Description
Green	Comments	Notes entered into program code. Set off from the code by a single quotation mark (').
Blue	Keywords	Any word or symbol recognized by E-Basic as part of the language.
Black	Statements	User-entered script.
Burgundy	Strings	Data values enclosed in quotation marks.

3.3 Using E-Run

3.3.1 Installation

The E-Run application may be installed using several installation options. Refer to Chapter 1-*Introduction* in the E-Prime User's Guide for a complete description of installation options. Before attempting to collect actual data, the E-Run installation should be tested.

3.3.1.1 Hardware Key

The E-Prime system is copy-protected, and requires the use of a hardware key that connects to the computer's parallel or USB port. The hardware key is required during installation and when using the development environment (i.e., E-Studio). A single user or multi-pack license permits



the user to develop experiments and analyze data with E-Prime, one computer at a time. However, the hardware key is not required for the run application (E-Run). E-Run may be installed and run on any number of machines in a single lab simultaneously. For a site license, the hardware key is required to be connected only during the installation, but not when working with E-Prime in any other capacity. For further information about the hardware key, refer to the User's Guide (Chapter 1-*Introduction*, section 1.4).

3.3.2 Compile versus Generate

In E-Studio's toolbar, clicking the Generate button (Figure 1) tells all E-Objects to convert their properties into E-Basic code. E-Studio gathers this information and adds its own to create the EBS file, which can be run using the E-Run application. The Compile command can be executed by clicking the Compile button on the toolbar in the E-Run application (Figure 2), or by pressing CTRL+F7. The Compile command takes the generated EBS file from E-Studio, and feeds it into the E-Basic compiler. The compilation process checks the syntax of the EBS file's contents to make sure that it is in an acceptable format to run. By default, E-Studio will compile after a successful generate.



Figure 1: Generate button within the E-Studio application.



Figure 2: Compile button within the E-Run application.

3.3.3 Errors

Please refer to Appendix A – *Error List* in this volume for a listing of design-time and run-time errors, and suggestions for resolving them.

3.3.3.1 Compile Errors

A compile error occurs during the process of compiling. It is not a logical error, but a syntax error (e.g., incorrect number of parameters, typing mistake, etc.). Refer to section 3.3.3.3 for examples of common compile errors. In E-Studio, a compile error can arise after clicking the Generate or the Run button (Figure 1), selecting Generate or Run in the E-Run menu (Figure 2), or pressing CTRL+F7 for Generate and F7 for Run.

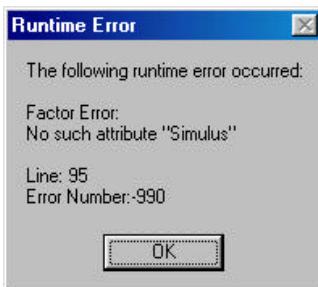
3.3.3.2 Run-Time Error

After completing the process of compiling, a run-time error is an error thrown during program execution when a line in the script cannot be completed. For specific examples of run-time errors, refer to the table in section 3.3.3.3. A run-time error may arise after the experiment has



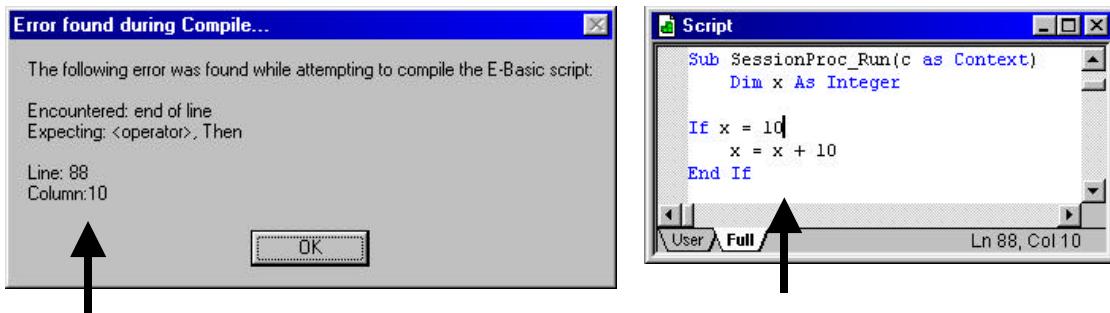
been launched and the startup information has been accepted. The experiment begins to run, but E-Run terminates when the error is encountered in the script. If running from E-Studio, an error dialog is displayed announcing the cause of the run-time error. If running from E-Run, E-Run displays an error dialog and returns to E-Run upon dismissal, with the cursor placed in the script at the line where the error occurred.

Spelling errors are the most common run-time errors in E-Studio. For example, the error below occurred due to the misspelling of an attribute (i.e., "Simulus" instead of "Stimulus").

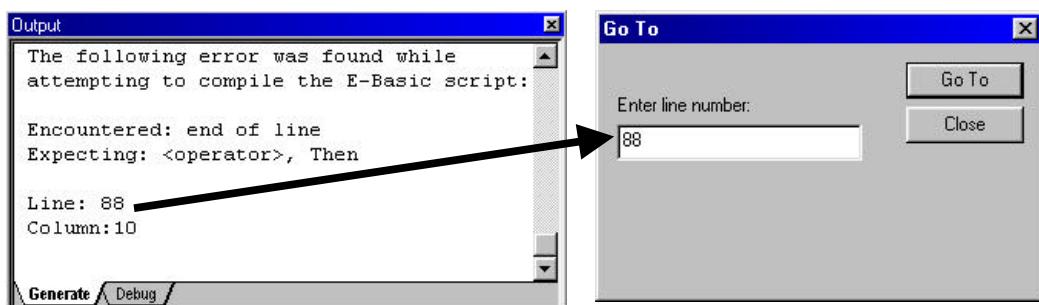


3.3.3.3 Locating Errors

When an error occurs, the error dialog displays the type of error (i.e., compile or runtime), the error description, and the location of the error in the script (i.e., line and column). If running from E-Studio, the Script window is automatically opened in E-Studio, and the cursor is placed at the line in the full script where the error was encountered.



To view an error message after the error dialog has been dismissed, display the Output window in E-Studio via the View menu. Compile errors are displayed on the Generate tab, and run-time errors are displayed on the Debug tab. To search for the specific line in the script, press CTRL+G to present the Go To dialog, and enter the Line number of the error in the field provided.





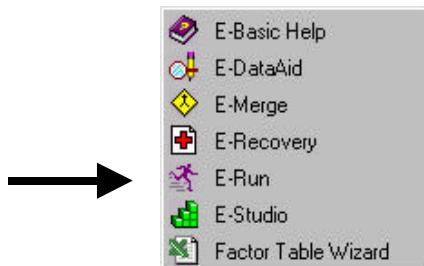
The output displayed on the Generate tab is overwritten during each successive generation. Output sent to the Debug tab is appended. To clear the Debug tab, right click in the Output window and select the Clear command.

3.3.4 Running Programs

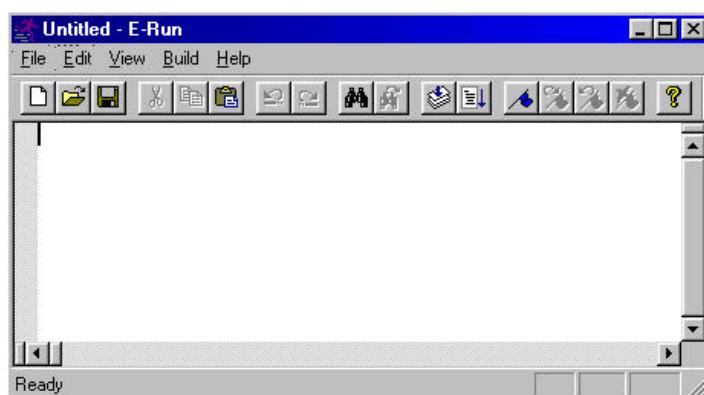
Once program development is completed, the program has been thoroughly tested, and the data file has been examined to determine that the necessary information is logging appropriately in E-Studio, data collection may begin. With a subject-ready program, E-Studio is no longer needed to launch the E-Run application, and data may be collected on multiple machines concurrently.

To run a completed program, the E-Basic Script (EBS) file should be launched directly from E-Run. EBS files, once completed, do not need to be run from E-Studio since the extra memory and additional files (i.e., EBS, ES, WNDPOS, EDAT and TXT) are not necessary for simply running a pre-created file. It should be noted that the generation of an E-Studio (ES) file to an E-Basic script (EBS) file is a one-way process. There is no way to generate an ES file from an EBS file.

When collecting data, all unnecessary applications (i.e., everything other than E-Run) should be closed. To launch E-Run, from the Start menu, choose Programs and E-Prime to display the E-Prime menu. From the E-Prime menu, select E-Run to launch this application.

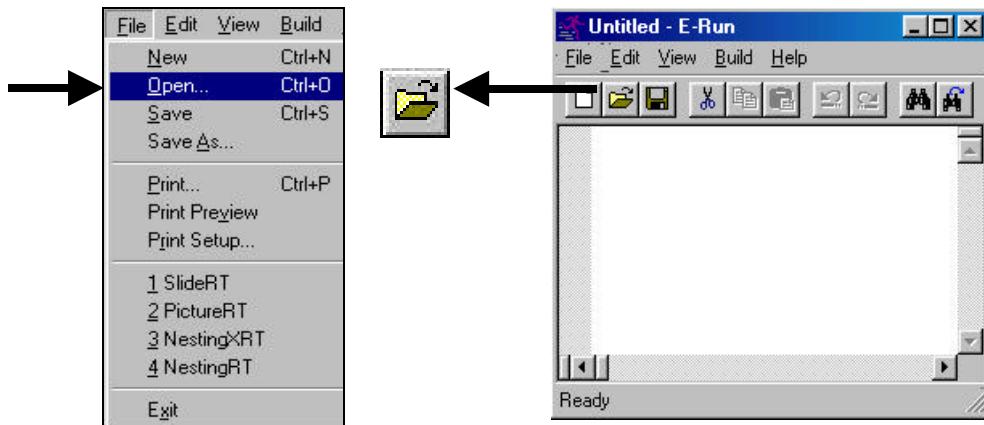


The E-Run application opens to a blank EBS file. While it is possible to either enter new script at this point or open an existing EBS file and edit script in the E-Run application window, it is not recommended that E-Run be used in this manner. The single most important reason for this is because the EBS file is regenerated every time a compile procedure is performed from within E-Studio. Any changes made to the EBS file from the E-Run application would be overwritten when the EBS file is regenerated from within E-Studio.

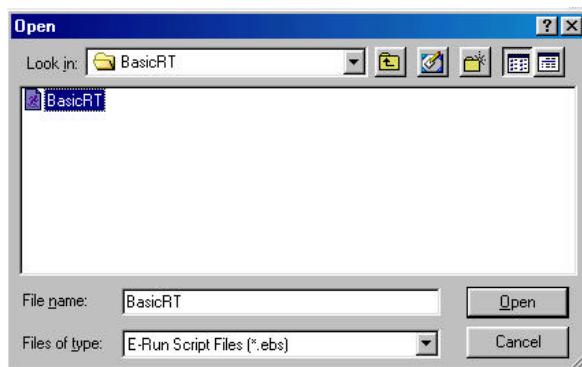




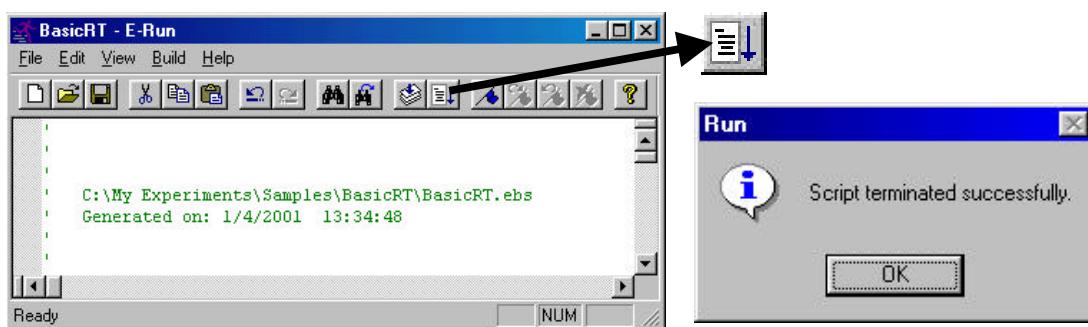
To open a pre-created EBS file in E-Run, select the Open command from the File menu or click the Open tool button.



The Open dialog will be displayed to allow the user to navigate to the appropriate directory and the file to be opened. Select the EBS file to be opened by clicking it, or type the name directly into the File Name field in the Open dialog, and click the Open button.



The generated E-Basic script for the selected experiment will be loaded into E-Run. To run the script, click the Run tool button, or simply press the F7 key. After the run is complete, E-Run will return to the window displaying the EBS file, and will display a dialog indicating the successful termination of the script. If errors occur, please refer to section 3.3.3 of this chapter for further assistance.





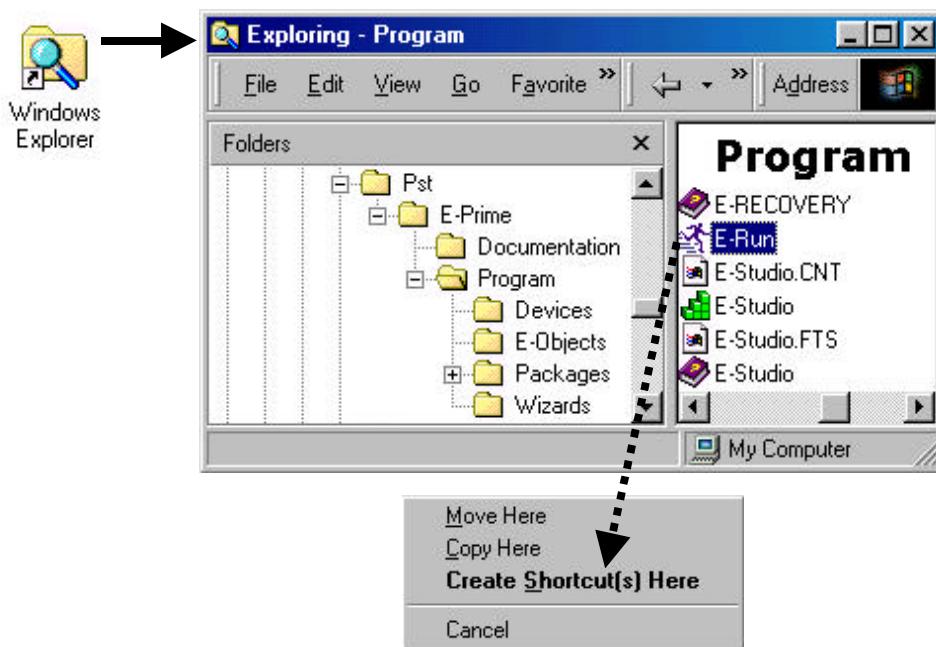
3.3.5 Sharing Pre-Developed Programs

In order to run an experiment on a machine other than the development machine, or to distribute an experiment to a colleague, each station must have at least the E-Run application installed and the EBS file. When necessary, auxiliary files such as sound files (which have the WAV extension), and image files (which have the BMP extension), must also be included. The user must copy the E-Prime Basic Script file (*.EBS) and any auxiliary files required by the program (e.g., BMP or WAV files) to all subject station machines. Refer to the User's Guide (Chapter 2, section 2.13.2) for further hints on distributing programs.

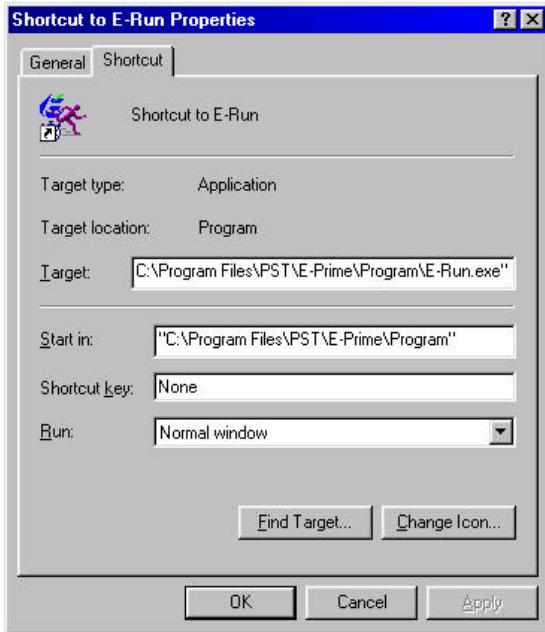
3.3.6 Create a Shortcut to an Experiment

A shortcut on the desktop allows the user to launch an experiment without having to go to its permanent location and load it into E-Run (i.e., navigate to either the E-Run or E-Studio applications and then load the desired EBS or ES file). Shortcuts are especially useful for experiments that are used frequently.

The first step in configuring an experiment shortcut is to open Windows Explorer. Next, navigate to the Program folder within the E-Prime installation (default installation is C:\Program Files\PST\E-Prime\Program). Right-click the icon for the E-Run.EXE application and drag it to the desktop. From the context menu, select the “Create Shortcut(s) Here” option.

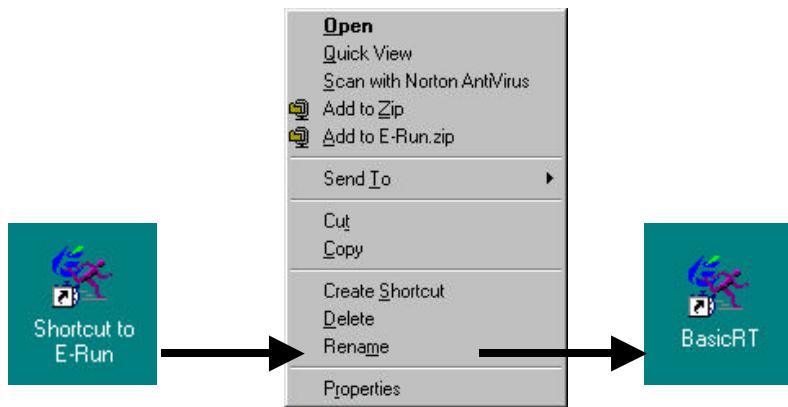


A shortcut to E-Run will be created on the desktop. Right click the shortcut to display the context menu, and select “Properties” to set the path of the shortcut. In the Shortcut to E-Run Properties dialog, the Target field will point to the location of E-Run (i.e., the default installation places E-Run.exe in the C:\Program Files\PST\E-Prime\Program folder).



Edit the “Target” field in the Shortcut to E-Run Properties dialog to include two switches and the name of the EBS file to be run. Edit the “Start in” field to contain the full path of the EBS file location. For example, to create a shortcut to the sample program BasicRT.ebs (assuming the default installation), the Target and Start in fields would be modified as follows (including all quotes, and matching case):

Target: “C:\Program Files\PST\E-Prime\Program\E-Run.exe” “/a” “/m” “BasicRT.ebs”.
Start in: “C:\My Experiments\Samples\BasicRT”.



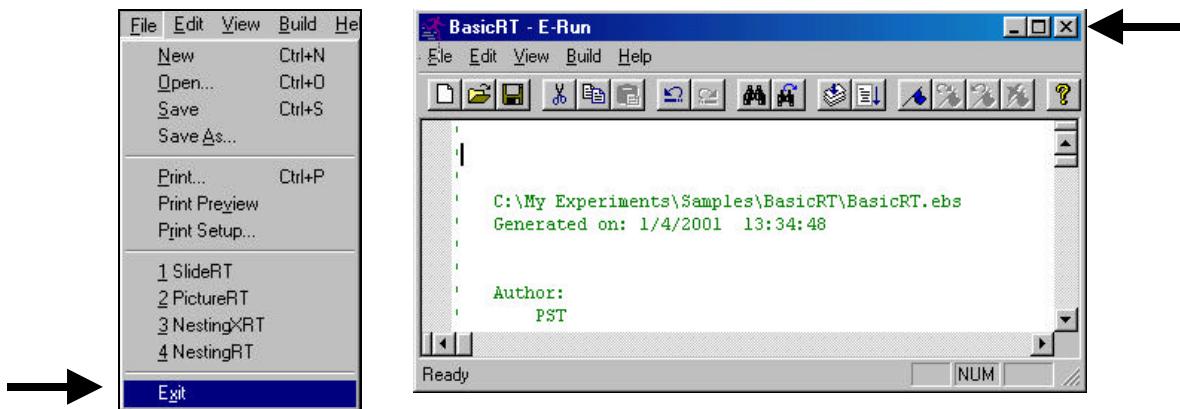
Data files will be created in the folder containing the EBS file. To rename the shortcut, right-click and select the “Rename” command from the context menu.

To launch the experiment from the shortcut, either double click the shortcut, or highlight it and press Enter. This will automatically open E-Run, load the EBS file, and compile and run the experiment.



3.3.7 Exiting E-Run

To exit E-Run, select the Exit command in the File menu, or click the Close button in the upper right corner of the display.





**E-Prime Reference Guide
Chapter 3: E-Run**

This page intentionally left blank.



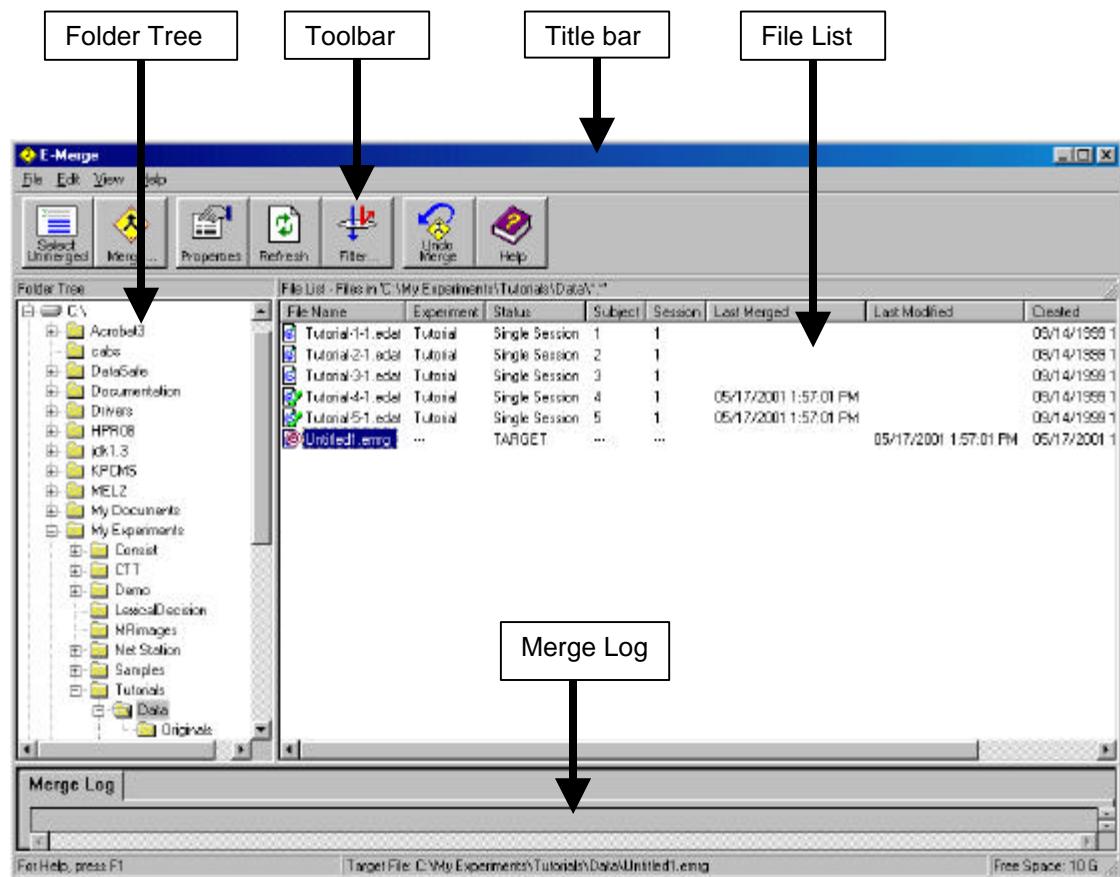
Chapter 4: E-Merge

4.1 About the E-Merge Application

The E-Merge application merges single session data files collected using E-Run into E-Prime merged data files. This application is used to merge single session data files into a master file for analysis.

4.2 About the Interface

The title bar is displayed at the top of the application screen, and displays the name of the application (i.e., E-Merge) as well as the E-Merge icon. The menu bar is displayed below the title bar. The menu bar lists the menus available within the E-Merge Application, including the File, Edit, View and Help menus (refer to section 4.2.1). The toolbar is displayed below the menu bar, and includes frequently used functions within E-Prime, including Select Unmerged, Merge, Properties, Refresh, Filter, Undo Merge, and Help. Other features include the Folder Tree view, File List view and Merge Log view.





4.2.1 Menus

4.2.1.1 File Menu

Commands	Shortcut	Function
<u>Open</u>	Ctrl+O	Opens the selected file using E-DataAid. Only one file may be opened at a time. If no file is selected, more than one file is selected, or an invalid file is selected, the Open command is disabled. Once opened in E-DataAid, the file may be modified, if necessary (e.g., renaming variables or changing subject numbers to avoid conflicts).
<u>Merge...</u>	Ctrl+M	Allows the combining of one or more data sessions into a target file. The Merge command is used to initiate the merge process. If no file is selected, or an invalid file is selected (i.e., non-E-Prime data file), this command is disabled.
<u>Set as Target...</u>	Ctrl+T	Sets the selected file as the target file for collection of data from individual source files during a merge operation. If no file is selected, or an invalid file is selected, this command is disabled.
<u>Properties</u>	Alt+Enter	Displays specific information about a selected file.
<u>Exit</u>		Closes the E-Merge application.

4.2.1.2 Edit Menu

Commands	Shortcut	Function
<u>Undo Merge</u>	Ctrl+Z	Cancels the last merge operation. E-Merge supports only one level of undo, so only the most current merge operation can be undone. If no merge operation has been performed, Undo Merge is disabled.
<u>Select Unmerged</u>	Ctrl+U	Selects (highlights) all single session data files in the File List view that have not been merged into another file.
<u>Select All</u>	Ctrl+A	Selects (highlights) all E-Prime data files in the File List view. The Select All command does not distinguish between merged, unmerged, and target files (i.e., all E-Prime data files are selected).

4.2.1.3 View Menu

Commands	Shortcut	Function
<u>Toolbar</u>		Toggles the display of the Toolbar on or off depending on its current state. A checkmark next to the Toolbar command indicates the Toolbar is displayed.
<u>Status Bar</u>		Toggles the display of the status bar on or off depending on its current state. A checkmark next to the Status Bar command indicates that the status bar is displayed.
<u>Quick Reference</u>		Toggles the display of the Quick Reference dialog on or off depending on its current state. The Quick Reference dialog is displayed by default when E-Merge is opened. This option may be changed by unchecking the initial display default for the Quick Reference dialog using the Options command under the View menu. A checkmark next to the Quick Reference Command indicates that the Quick Reference dialog is displayed.
<u>Refresh</u>	F5	Updates the File List view for the folder currently opened in the Folder Tree view.
<u>Filter...</u>		Displays a dialog allowing the File List view to be filtered according to a designated pattern (i.e., file name). A filter pattern may be entered by typing directly in the "Select a filter" field on the Filter on File Name dialog, or by choosing one of the available choices from the drop-down box. The default filter pattern is "All Files (*.*)" to display all files within the current folder. Other available patterns include only single session data files (*.EDAT), only merged data files (*.EMRG), and all E-Prime data files (*.EDAT, *.EMRG). Any filter pattern entered by the user will be added to the drop-down box as an available pattern until the application is closed.



View Menu continued...

Commands	Shortcut	Function
<u>Options...</u>		Displays a dialog allowing the setting of the initial display options (i.e., when E-Merge is opened), and the setting of an option for the display and handling of conflicts during merge operations.
<u>Folder Tree</u>	Alt+0	Activates the Folder Tree view (i.e., places the focus in this view so that any action taken, such as moving the cursor, would occur within this view).
<u>File List</u>	Alt+1	Activates the File List view (i.e., places the focus in this view so that any action taken, such as moving the cursor, would occur within this view).
<u>Merge Log</u>	Alt+2	Activates the Merge Log view (i.e., places the focus in this view so that any action taken, such as moving the cursor, would occur within this view).

4.2.1.4 Help Menu

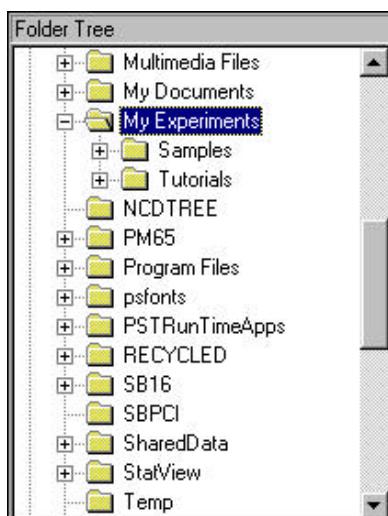
Commands	Shortcut	Function
<u>Help Topics</u>	F1	Displays the application's Help system.
<u>About E-Merge...</u>		Displays a dialog containing the application's version information.

4.2.1.5 Context Menu

A context menu may be accessed by right clicking a file name in the File List view. The context menu includes the Open, Set As Target, Merge, and Properties commands. The context menu simply permits quick access to the commands frequently applied to the items in the File List view.

4.2.2 Folder Tree View

The Folder Tree view is displayed on the left side of the application display below the toolbar. The Folder Tree view displays a hierarchical list of drives and folders available on the machine. Navigate to a particular drive or directory using the mouse or arrow keys, and click on a specific folder to open it. A folder may be expanded by clicking on the "+" next to the folder.





4.2.3 File List View

The File List view is displayed on the right side of the application display below the toolbar. The File List view displays the names of the individual files contained within the folder opened in the Folder Tree view. Columns may be sorted in ascending or descending order by clicking on the column header. Specific files may be selected by clicking on the name of the file in the File Name column. The File List view includes information specific to the individual files such as the size of the file, and information pertaining to the creation, modification, or merge status of the file.

File Name	Experiment	Status	Subject	Session	Last Merged	Last Modified	Created	Size in KB
MergedData.emrg	---	TARGET	---	---		2/3/00 2:21:38 PM	2/3/00 2:21:38 PM	18
Tutorial-1-1.edat	Tutorial	Single Session	1	1	2/3/00 2:21:38 PM		9/14/99 1:37:45 PM	8
Tutorial-2-1.edat	Tutorial	Single Session	2	1	2/3/00 2:21:38 PM		9/14/99 1:38:39 PM	8
Tutorial-3-1.edat	Tutorial	Single Session	3	1	2/3/00 2:21:38 PM		9/14/99 1:39:31 PM	8
Tutorial-4-1.edat	Tutorial	Single Session	4	1	2/3/00 2:21:38 PM		9/14/99 1:40:22 PM	8
Tutorial-5-1.edat	Tutorial	Single Session	5	1	2/3/00 2:21:38 PM		9/14/99 1:41:15 PM	8

4.2.3.1 Icons

To the left of the filename for each file in the File List view is an icon indicating the status of the file.

	Unmerged, unaltered E-Prime data file
	Unmerged E-Prime data file containing altered data.
	E-Prime data file that has been merged to another file.
	E-Prime data file that has been merged, and contains altered data.
	E-Prime data file selected as target file.
	E-Prime data file containing more than one session of data.
	All non-E-Prime data files.

4.2.3.2 File Name

The File Name column displays the filename without the path for each file contained within the folder selected in the Folder Tree view.

4.2.3.3 Experiment

The Experiment column lists the name of the experiment with which the data file is associated. For merged data files, the Experiment column displays dashes, because data from different experiments may be contained within the same file. This column is left empty for all non-E-Prime data files.



4.2.3.4 Status

The Status column contains one of three values for E-Prime files: Single Session, Merged data, or Target. The Status column remains blank for non-E-Prime files.

Single Session

Single session files are E-Prime data files generated by the E-Run application, which contain data from a single session. Single Session files have the EDAT file extension.

Merged Data

Merged data files are E-Prime data files which contain data from more than one session, and which are not designated as the target file. Merged data files have the EMRG extension. Merged data files may be merged into other files (i.e., act as a source file) or receive data from other files (i.e., act as a target file).

Target File

The **target file** is the file into which sessions of data are to be merged during a particular merge operation. Single session files may not be named as target files because data may not be merged into them. Target files may or may not be located in the same directory as the source files, and may or may not exist prior to the merge operation. If the target does not exist prior to the merge operation, it will be created by E-Merge during the operation.

If the file contains modifications to the data, three asterisks are added to the Status column value. When a source file containing data alterations is merged into a target file, E-Merge carries all data alteration flags from the source file to the target file. Therefore, when merging a source file with data alterations into a target file without data alterations, the target file will contain data alterations at the conclusion of the merge operation. After the merge operation is completed, the Status column value for the target file will be updated to include “***”, indicating altered data.

4.2.3.5 Subject

The Subject column displays the subject number assigned within an E-Prime data file. For merged E-Prime data files, this column displays three dashes because merged data files may contain data for more than a single subject. For files other than E-Prime data files, this column remains empty.

4.2.3.6 Session

The Session column displays the session number assigned within an E-Prime data file. For merged E-Prime data files, this column displays dashes because merged data files may contain more than a single session of data. For files other than E-Prime data files, this column remains empty.

4.2.3.7 Last Merged

The Last Merged column contains the date the file was last merged into another file. This column is blank if the file was never merged into another file or if the file is not an E-Prime data file.

4.2.3.8 Last Modified

The Last Modified column displays the last date any changes were made to the file. The column remains blank if the file was never modified or if the file is not an E-Prime data file. Modifications



to the file need not include altering the data. For example, saving a new comment to a file will change the Last Modified date but will not alter the data.

4.2.3.9 Created

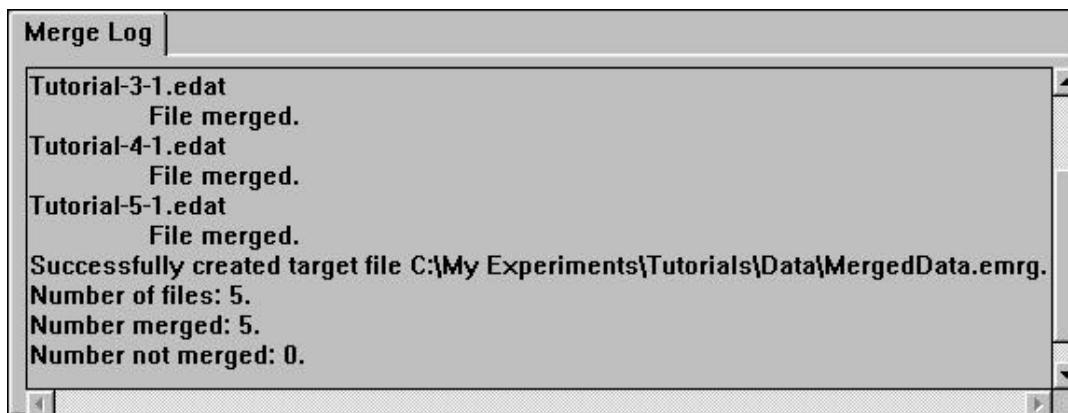
The Created column displays the creation date for E-Prime data files, and remains blank for all other files. The creation date refers to the date a single session data file was generated by E-Run, or the date a merged data file was created by E-Merge during a merge operation.

4.2.3.10 Size in KB

The Size in KB column displays the size of the file in kilobytes.

4.2.4 Merge Log View

The Merge Log view is displayed at the bottom of the application display. The Merge Log view displays detailed results of the latest merge operation performed. The results of each merge operation overwrite the results of previous merge operations in the Merge Log view. The Merge Log can contain up to 30,000 characters. The scroll bars may be used to view portions of the data not within the display, and the results from the Merge Log may be copied and pasted to a text document in another application (e.g., to save previous merge operation results).



4.2.5 Status Bar

The status bar is displayed at the bottom of the application display below the Merge Log view, and is divided into three sections.

4.2.5.1 Status

The status pane, on the far left, displays the status of the application (e.g., displays “Merging” during a merge operation, or displays the function of a command on which the cursor is placed in a menu).





4.2.5.2 Target File

The Target File pane, in the middle of the status bar, lists the full path and name of the file currently designated as the target. If no target has been selected, the Target File pane lists the name of the target file as "Untitled"

Target File: C:\My Experiments\Tutorials\Data\MergedData.emrg

4.2.5.3 Free Space

The Free Space pane on the far right displays the drive space available for the files in the File List view.

Free Space: 10 GB

4.3 Using E-Merge

4.3.1 Annotations

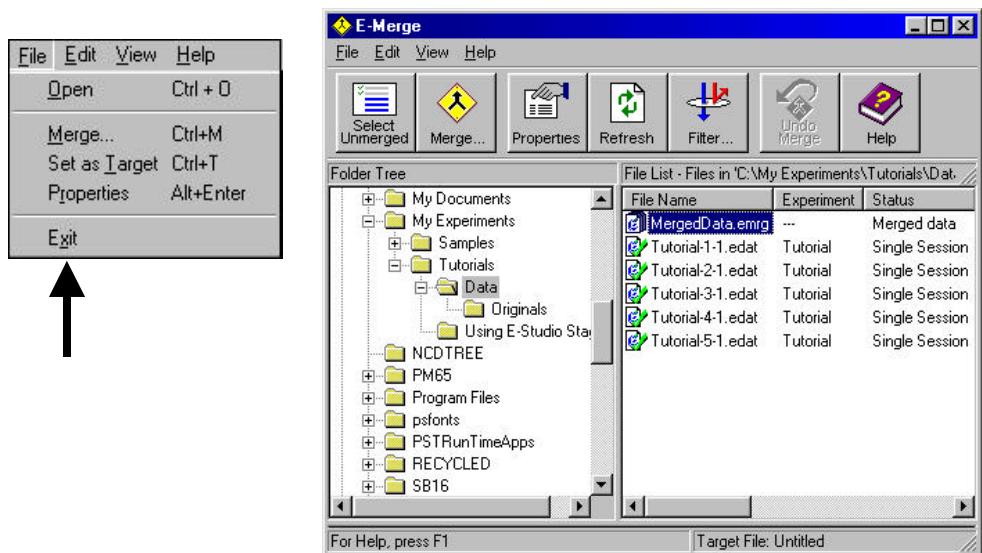
Annotations are comments or short descriptions of actions performed on a file, such as merging or editing. For example, if a source file is merged into a target file, an annotation is written to the source file indicating the file into which the session of data was merged. This information may be retrieved by examining the source file's merge output history, which is accessed by double-clicking the file name and selecting the Merge Output History tab from the Properties dialog. In addition, E-Merge writes an annotation to the target file indicating the source from which the data was merged. This information may be retrieved by examining the target file's merge input history, which is accessed by double-clicking the file name and selecting the Merge Input History tab from the Properties dialog.

Annotations may be stored at the session level (as is the case with annotations relating to merge operations) or globally (as is the case with annotations relating to the renaming of variables). In most cases when a session is merged from a source file to a target file, the annotation is merged as well. When a source file contains global annotations, the global annotations are merged into the target file during the merge operation. In the target file, E-Merge appends the source file name in parentheses to the end of the global annotation being merged. This alerts the user that the global annotation originated in another file.

The exception to this involves annotations relating to merge operations. While an annotation for each merge operation is written to both the source file and the target file (as described above), annotations concerning previous merge operations are not carried from the source file to the target file during subsequent merge operations. The reason for this is that, in most cases, information concerning previous merge operations is relevant only to the source file, and therefore it is not necessary to record it in the target file.

4.3.2 Exiting E-Merge

To exit E-Merge, select the Exit command in the File menu, or click the close button in the upper right corner of the display.



4.3.3 Conflicts During Merge Operations

The E-Merge application automatically checks for potential data file conflicts and presents options for dealing with any conflicts encountered. By default, the merge process will be halted when a conflict is encountered and will prompt the user for an action. The action to be taken when conflicts are encountered may also be determined prior to the merge operation using the Options command under the View menu. In all cases, the application will display any conflicts encountered and subsequent actions in the Merge Log view at the end of the merge operation. Possible conflicts are described in the following sections.

Conflict 1 - The session contained in a source file has the same subject number, session number, and experiment name as a session in the target file.



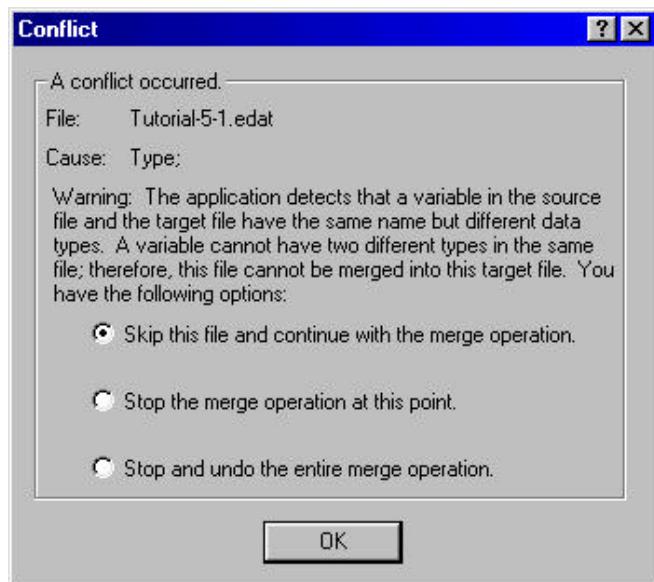


E-Run assigns a unique internal identifier to each session of data. Although the user is never aware of the identifier, the E-Merge application internally uses this identifier to make sure that two different runs of an experiment were not given the same subject and session number. For example, if two subjects were mistakenly assigned the same subject and session number, the files would look as if they were duplicates, but the data would be different.

If E-Merge detects this conflict: 1) skip the file causing the conflict and continue with the merge operation, 2) continue with the merge operation to merge the file causing the conflict into the target file, 3) stop the merge operation at the point at which the conflict occurred, or 4) negate the current merge operation. If the merge operation is stopped at the point of conflict, all files merged prior to the occurrence of the conflict remain merged in the target file. If the merge operation is negated, all files merged prior to the conflict are unmerged.

If the file causing the conflict is merged into the target file, the target file should be edited using E-DataAid so no two sessions with the same subject number, session number, and experiment name exist within the merged data file. If the file causing the conflict is not merged into the target file, either the source file or the target file may be edited using E-DataAid to modify the duplicate subject number, session number, or experiment name, and the merge operation may be performed again.

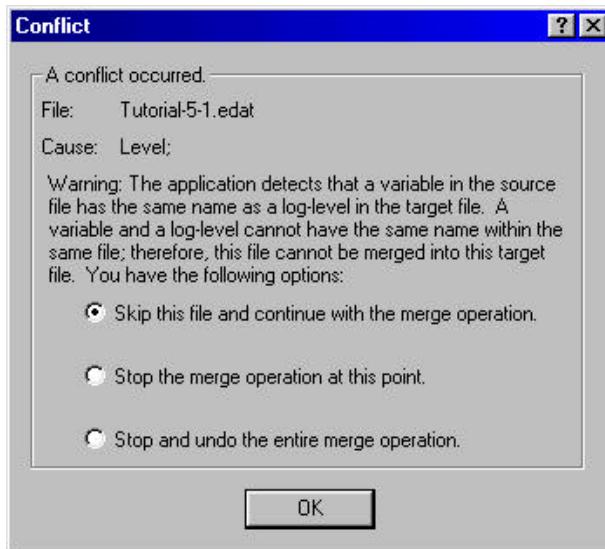
Conflict 2 - A variable in the source file has the same name as a variable in the target file, but a different data type.



A variable may not have two different data types within the same data file. If this conflict occurs: 1) skip the file causing the conflict and continue with the merge operation, 2) stop the merge operation at the point at which the conflict occurred, or 3) negate the current merge operation. If the merge operation is stopped at the point of conflict, all files merged prior to the occurrence of the conflict remain merged in the target file. If the merge operation is negated, all files merged prior to the conflict are unmerged. The variable name causing the conflict may be edited in either the source file or the target file using E-DataAid, and the merge operation may be performed again.

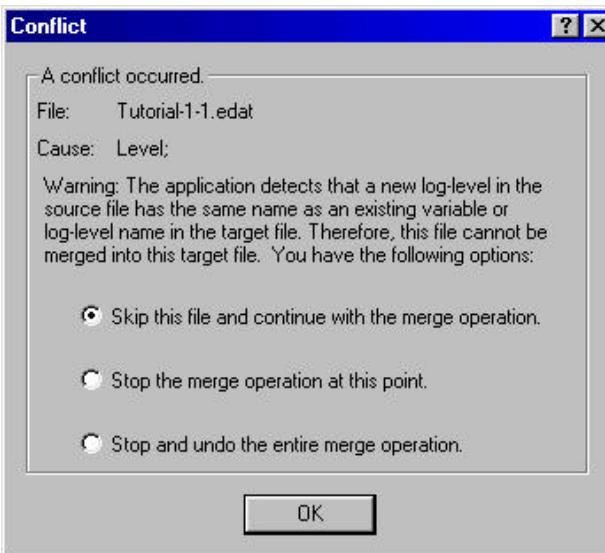


Conflict 3 - A variable in the source file has the same name as a log level in the target file.



A variable and a log level may not have the same name within a data file. If this conflict occurs: 1) skip the file causing the conflict and continue with the merge operation, 2) stop the merge operation at the point at which the conflict occurred, or 3) negate the current merge operation. If the merge operation is stopped at the point of conflict, all files merged prior to the occurrence of the conflict remain merged in the target file. If the merge operation is negated, all files merged prior to the conflict are unmerged. Either the variable name in the source file or the log level in the target file may be edited using E-DataAid to eliminate the conflict, and the merge operation may be performed again.

Conflict 4 - A log level in the source file has the same name as a different log level in the target file.

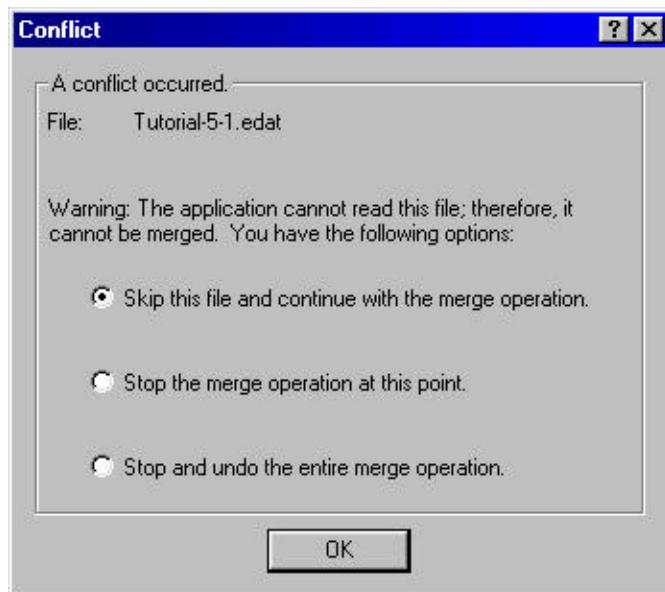




Two logging levels may not have the same name in the same data file. If this conflict occurs: 1) skip the file causing the conflict and continue with the merge operation, 2) stop the merge operation at the point at which the conflict occurred, or 3) negate the current merge operation. If the merge operation is stopped at the point of conflict, all files merged prior to the occurrence of the conflict remain merged in the target file. If the merge operation is negated, all files merged prior to the conflict are unmerged. Using E-DataAid, the duplicate log level name may be edited in either the source file or the target file, and the merge operation may be performed again.

Conflict 5 - The source file cannot be read.

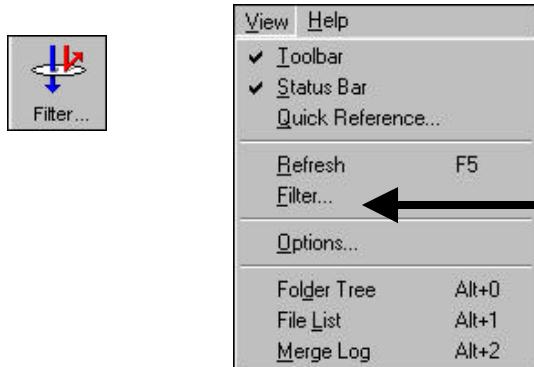
This conflict may occur if the source file is currently opened in another application, or if the file is unreadable. If this conflict occurs: 1) skip the file causing the conflict and continue with the merge operation, 2) stop the merge operation at the point at which the conflict occurred, or 3) negate the current merge operation. If the merge operation is stopped at the point of conflict, all files merged prior to the occurrence of the conflict remain merged in the target file. If the merge operation is negated, all files merged prior to the conflict are unmerged.



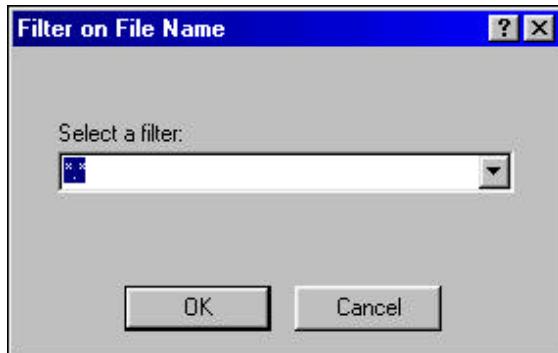
If the file is opened in another application (e.g., E-DataAid), close the other application and the merge operation may be performed again.

4.3.4 Filtering the File List View

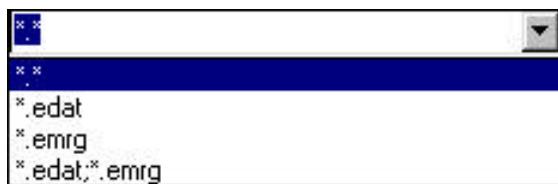
The set of files displayed in the File List view may be filtered according to file name to limit the number of files listed. The filter is applied to include all files fitting a designated pattern. The Filter on File Name dialog may be displayed using the Filter tool button, or the Filter option under the View menu.



Four filters are supplied in the Filter on File Name dialog: all files (*.*), single session files (*.EDAT), merged data files (*.EMRG), and all E-Prime data files (*.EDAT; *.EMRG). The default setting displays all files (*.*).



The default may be changed by typing directly in the Select a Filter field, or by choosing an option from the drop-down menu. Any filter pattern entered by the user is added to the list of available filters in the drop-down box until the application has been exited. Multiple filters may be entered using a semi-colon as a separator. Note: **Do not include spaces in addition to the semi-colon separating filters.**



Because the file name filter examines both long filenames and aliases during the search, some wildcard searches may yield unexpected results. For example, if the system has generated the alias *LONGFI~1* for *LongFileName*, a filter on a *1 pattern would return *LongFileName* even though the actual file name does not contain a 1.

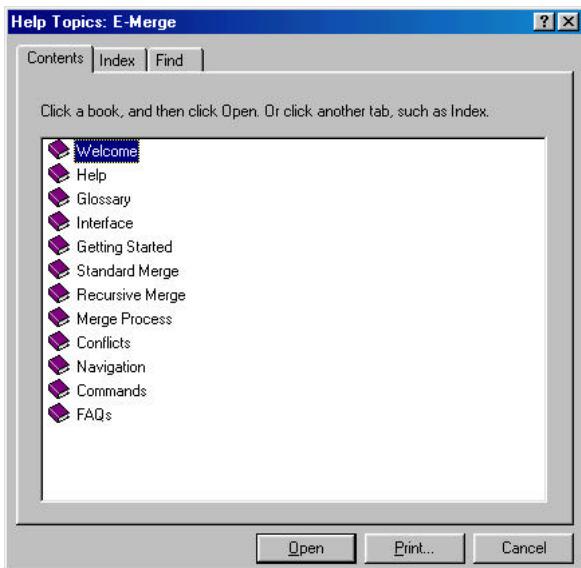
4.3.5 Help

The main Help system may be displayed using the **F1** key. The Help contents may be searched using an index listing of the available topics, or by searching for specific words or phrases in the Help topics. Context sensitive Help is also available.



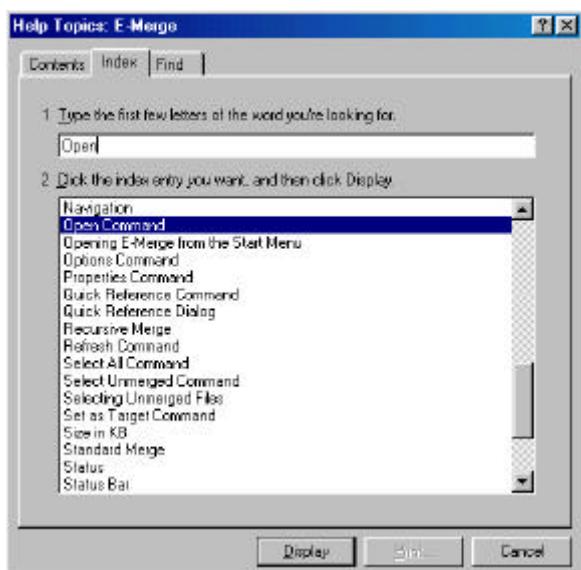
4.3.5.1 Contents

The Contents tab lists a table of contents for the Help topics available within the E-Merge Help system. The main Help topics are displayed as books, which may be opened to display related subtopics.



4.3.5.2 Index

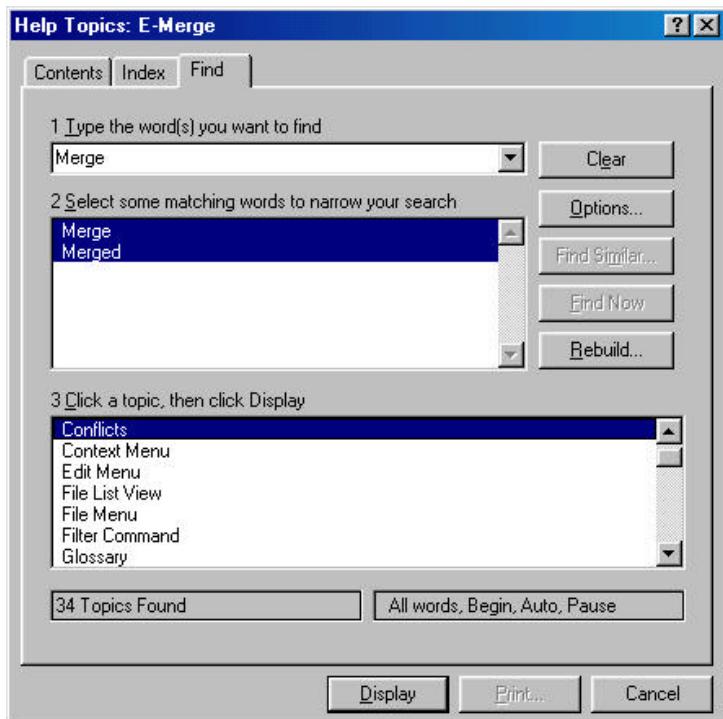
The Index tab displays an alphabetical listing of topics and commands within E-Merge. The Help information for a particular topic may be displayed by selecting the topic and clicking the Display button, or by double clicking the topic. The topics may be searched by typing directly in the first field of the Help Topics dialog, or by scrolling through the topics contained in the index using the scroll bar on the right side of the Help Topics dialog.





4.3.5.3 Find

The Find tab allows searching for specific words or phrases within the Help topics. After typing in a word to search for in the Help topics, the topics in which the word appears are listed, as well as additional suggestions to narrow the search. The Help information for a particular topic may be displayed by selecting the topic and clicking the Display button, or by double clicking the topic.



4.3.5.4 Context Sensitive Help

Context sensitive Help is available by various methods. Context sensitive Help may be invoked by clicking the **?** button in the title bar of the active dialog box, and then clicking the item to be queried. Another method is to place the cursor over the point to be queried in the dialog box, right click, and then click "What's This?" Finally, context sensitive Help may be invoked by placing the cursor over the item to be queried in the dialog box and pressing **F1**.





4.3.6 Icons

E-Merge uses icons in the File List view to reflect the state of each file.

Icon	Meaning
	E-Prime data file that has not been altered or merged to another file.
	E-Prime data file that has not been merged to another file, and contains altered data.
	E-Prime data file that has been merged to another file.
	E-Prime data file that has been merged to another file, and contains altered data.
	E-Prime data file selected as target file.
	E-Prime merged data file containing more than one session of data.
	All non-E-Prime data files.

4.3.7 Merging Files

E-Prime data files may be merged to create E-Prime **merged data files**. When data from more than one session of data is combined into a single file, it is referred to as a merged data file. The process of merging involves selecting one or more E-Prime data files and performing a **merge operation** to combine the data from those files into a **target file**. Both **single session data files** and merged data files may be merged into other files (i.e., may be selected as source files). Single session data files may not be named as target files because they may contain only one session of data (i.e., data may not merged into them).

4.3.7.1 Selecting Source Files

A single file may be selected as a source file by simply clicking on the file name in the File List view to highlight it. Multiple files may be selected as source files concurrently by several methods described below.

Select Unmerged

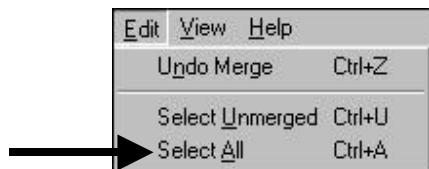
The Select Unmerged tool button or the Select Unmerged command under the Edit menu will highlight all E-Prime data files in the File List view which have not been merged into another file.





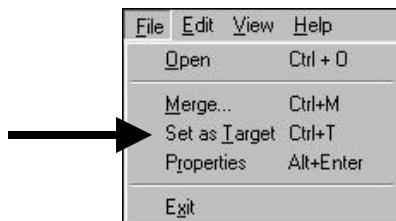
Select All

The Select All command under the Edit menu will select all E-Prime data files (i.e., merged and unmerged, single session and merged data files).

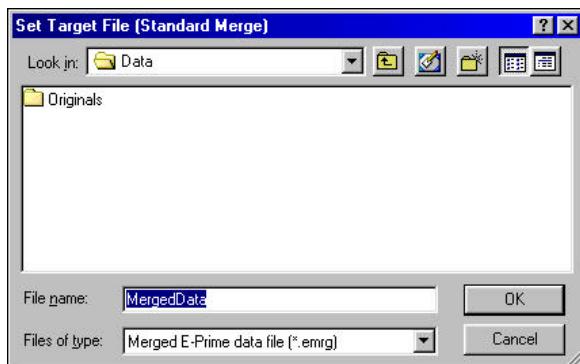


4.3.7.2 Selecting Target Files

In order for a merge operation to take place, a **target file** must be designated as the destination for the source data. A target file may be set prior to a merge operation by clicking on an eligible file to select it, and choosing Set as Target under the File menu.



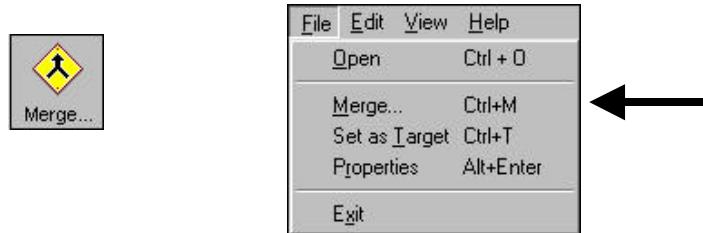
Eligible target files include existing E-Prime merged data files, or merged data files (EMRG) created during a merge operation using the Set Target dialog. If the target file is set prior to the merge operation, the Set Target File dialog appears as a confirmation of the chosen target file.



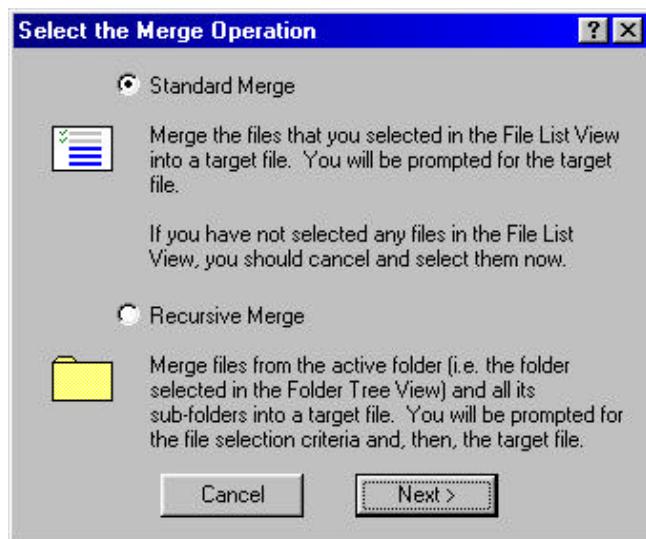
Note: If the target file is in a different folder than the active folder, the merge operation will require the user to navigate to the target even though it may have previously been set as the target. This is viewed as a necessary inconvenience to prevent accidental merging to the wrong target file.

4.3.7.3 Merge Operations

A merge operation involves the combining of data from one or more source files into a target file. A merge operation may be initiated using the Merge tool button on the toolbar, or by choosing the Merge option under the File menu.

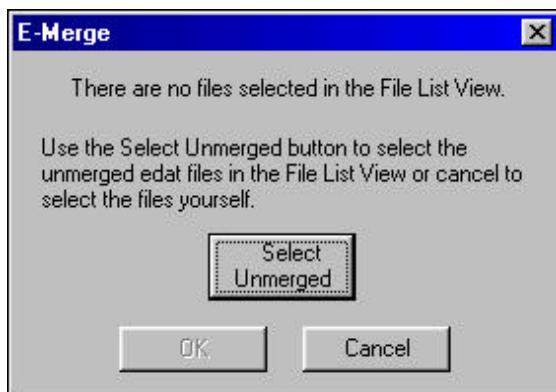


When a **merge operation** is initiated, a dialog is displayed which prompts the user to select the type of merge operation. Two types of merge operations are possible.



Standard Merge

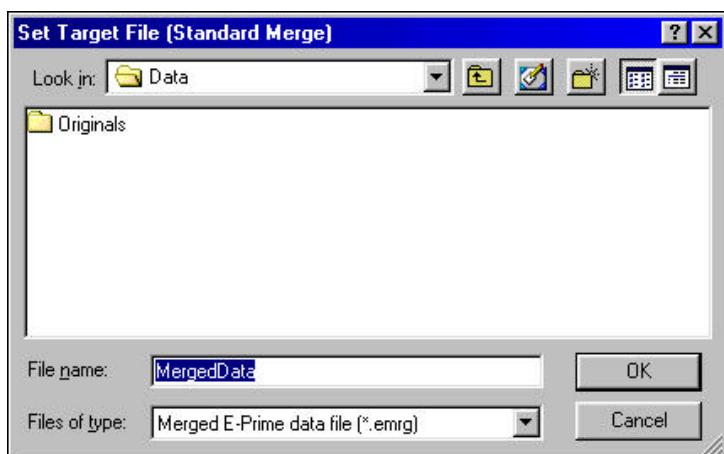
The standard merge operation (default selection) is the process of selecting source files from the File List view and merging them into a target file. A standard merge requires the source file(s) to be selected before the merge operation may occur. If at least one source file is not selected prior to the standard merge operation, a dialog is displayed indicating that no files have been selected (i.e., there is no data to merge).



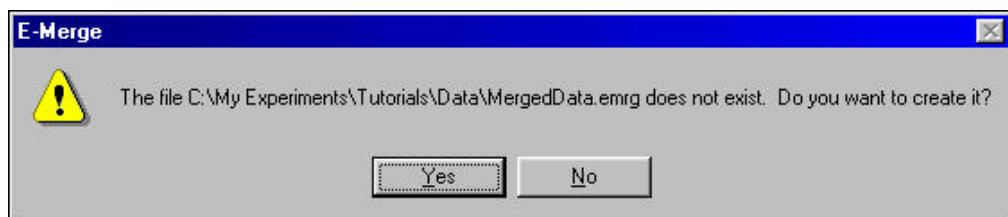


At this point, the user is given the option of selecting all unmerged data files within the current folder, or cancelling the merge operation. If the operation is cancelled, the user is returned to the File List view within E-Merge, where source files may be selected manually and the merge operation may be reattempted. If the Select Unmerged option is chosen, all unmerged single session E-Prime data files are selected (i.e., they become highlighted in the File List view). Once source files are selected, the OK button becomes active, which allows the user to continue with the merge operation.

The next step in any merge operation is to designate or verify the target file into which the data from the source file or files will be merged. The Set Target File dialog is displayed, allowing the user to navigate to the folder containing an existing target file, or to determine the folder into which a new target file will be placed after it is created.



If the target file designated in the Set Target File dialog exists, E-Merge simply reads the file and proceeds with the merge operation. If the named target file does not exist, the user is prompted with a dialog asking whether or not to create the file.

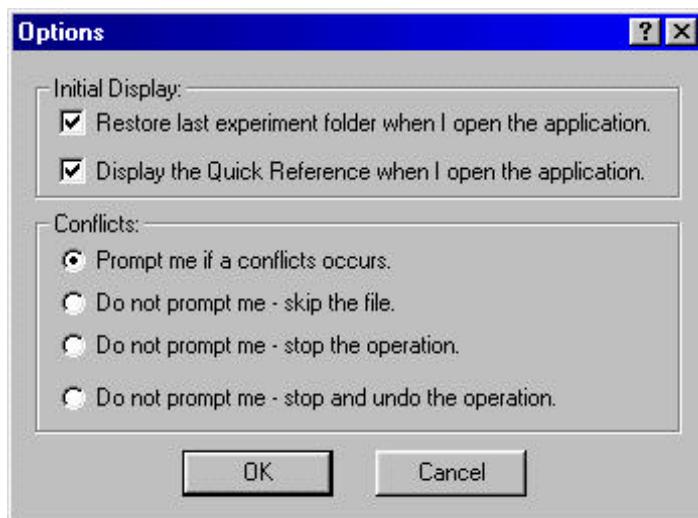


If the user chooses not to create the target file (i.e., selects “no”), the merge operation returns to the Set Target File dialog, and another name may be entered for the target file. If “yes” is selected, the file is created in the directory named within the Set Target File dialog, and the merge operation proceeds.

After the target file is selected and verified, E-Merge checks for low disk space. If disk space is low, a warning is displayed and the user is given the opportunity to cancel the merge operation. If disk space is adequate, E-Merge saves the information necessary to later undo the merge operation. If a merge operation cannot be undone, a warning is displayed prior to the merge operation, and the merge operation may be cancelled. E-Merge supports only one level of undo. Therefore, only the most current merge operation may be undone.



E-Merge then attempts to merge the selected source file(s) into the named target file. Each source file is examined for potential conflicts with the target file (refer to section 4.3.3 for conflicts). If a conflict occurs, an action must be chosen in order to continue with the merge operation. By default, a Conflict dialog will be displayed for each conflict encountered, and a course of action must be chosen for each conflict individually. Alternatively, the course of action may be set generally for all conflicts prior to the merge operation using the Options command under the View menu. By setting the course of action using this method, the Conflict dialog will not appear when a conflict occurs, and the same course of action will be taken for each conflict encountered.



The options available when setting the course of action prior to the merge operation allow the user to: 1) skip the file causing the conflict and continue with the merge operation, 2) stop the merge operation at the point at which the conflict occurred, or 3) negate the current merge operation. If the merge operation is stopped at the point of conflict, all files merged prior to the occurrence of the conflict remain merged in the target file. If the merge operation is negated, all files merged prior to the conflict are unmerged.

After checking for conflicts, E-Merge merges the data sessions from the source file(s) into the target file. E-Merge updates each source file by writing an annotation to it indicating the name of the target file into which the source file was merged. This information, useful for tracking the files into which the source file data was merged, may be retrieved by selecting a source file and viewing its Merge Output History (refer to section 4.3.7.7 for Merge Output History).

After the application merges all source files, E-Merge updates or writes the target file to the disk. In addition, E-Merge writes an annotation to the target file indicating the name of the source file(s) merged into the target file. This information, useful for tracking which source files were merged into a target file, may be retrieved by selecting a target file and viewing its Merge Input History (refer to section 4.3.7.6 for Merge Input History).

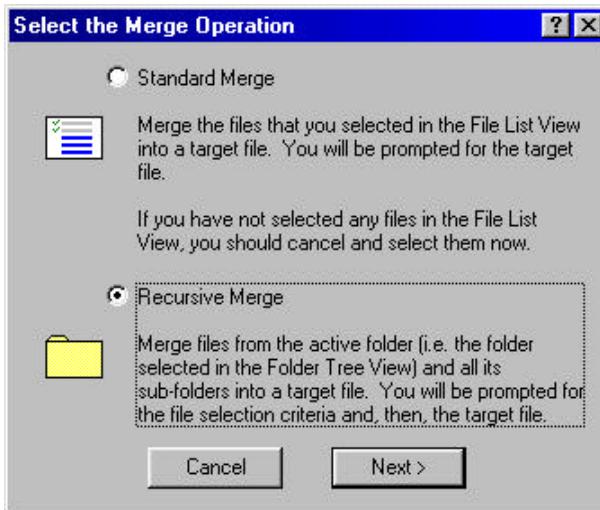
Recursive Merge

A recursive merge operation is the process of merging files that reside in the active folder and its sub-folders into a target file. The active folder is the folder selected in the Folder Tree view. The recursive merge differs from the standard merge in two ways. First, the recursive merge does not require the selection of individual files in the File List view. The files to be included in the recursive merge are designated during the merge operation via a dialog. Secondly, the recursive

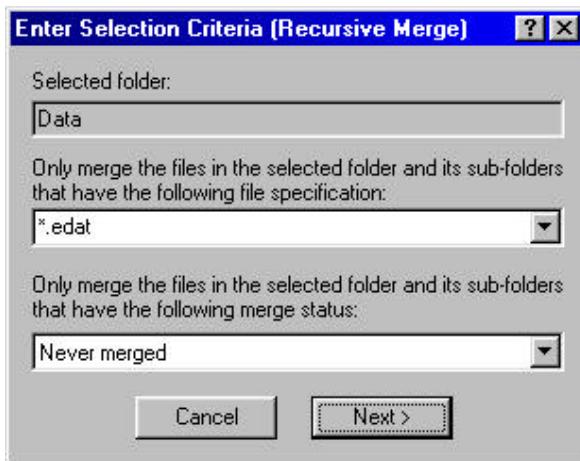


merge operation is not limited to the active folder. The recursive merge operation looks for all files matching the file specification in the active folder and all of its sub-folders.

Like the standard merge, a recursive merge operation may be initiated using the Merge tool button on the toolbar or by choosing the Merge option under the File menu. When asked to select the type of merge operation, click the radio button next to Recursive Merge.



When a recursive merge is run, it is necessary to designate the types of files in the selected folder and sub-folders to merge. The second step of the recursive merge displays the Enter Selection Criteria dialog, which permits the user to set the types of files to be included in the merge operation.

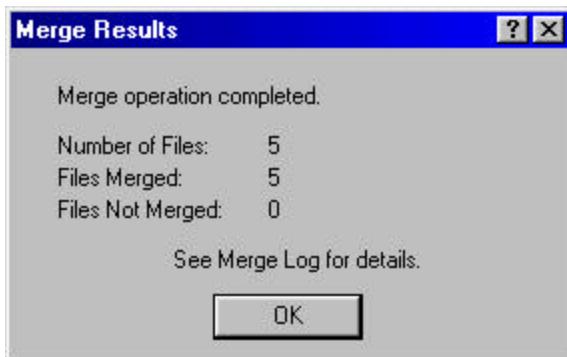


A recursive merge may include EDAT files, EMRG files (or both), and may include files that have never been merged, or were previously merged to a target (or both). Once the criteria are established for the recursive merge, the merge operation proceeds as the standard merge would, with the designation of the target file.

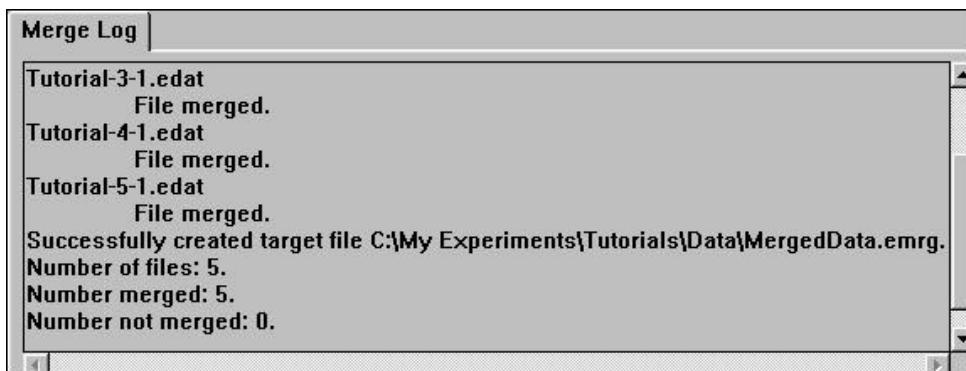


4.3.7.4 Merge Results

Upon completion of a merge operation, a Merge Results dialog is displayed listing: 1) the number of files E-Merge attempted to merge into the target file during the current merge operation, 2) the number of those files successfully merged, and 3) the number of files which were not successfully merged into the target file.



In addition, the Merge Log view lists a detailed summary of the merge operation, including: 1) the number of source files E-Merge attempted to merge into the target file during the current operation, 2) the number of those files successfully merged, and 3) the number of files which were not successfully merged into the target file. The Merge Log view lists each action performed during a merge operation. If a target file is created, a message indicating the creation of the file and the name of the file (including the entire file path) are displayed in the Merge Log view.



If an existing target file is named in the operation, a message is displayed indicating the target file has been updated.

Successfully updated target file C:\My Experiments\Tutorials\Data\MergedData.emrg.

Each source file used in the merge operation is listed in the Merge Log view, with a description of any conflict or warning encountered while that file was being merged and the course of action taken as a result of the conflict. If no conflict is encountered, the Merge Log view simply displays a "File Merged" message.

After each merge operation, the columns in the File List view are updated according to the results of the merge operation. The Status column lists the current status of each E-Prime data file as either Single Session (for *.EDAT files) or Merged Data (for *.EMRG files). When a merged data



file is set as the target for a merge operation, the status for that file is changed to TARGET until the E-Merge application is closed or until another target is set.

File Name	Experiment	Status	Subject	Session	Last Merged	Last Modified	Created	Size in KB
MergedData.emrg	---	TARGET	---	---		2/3/00 2:21:38 PM	2/3/00 2:21:38 PM	18
Tutorial-1-1.edat	Tutorial	Single Session	1	1	2/3/00 2:21:38 PM	9/14/99 1:37:45 PM	8	
Tutorial-2-1.edat	Tutorial	Single Session	2	1	2/3/00 2:21:38 PM	9/14/99 1:38:39 PM	8	
Tutorial-3-1.edat	Tutorial	Single Session	3	1	2/3/00 2:21:38 PM	9/14/99 1:39:31 PM	8	
Tutorial-4-1.edat	Tutorial	Single Session	4	1	2/3/00 2:21:38 PM	9/14/99 1:40:22 PM	8	
Tutorial-5-1.edat	Tutorial	Single Session	5	1	2/3/00 2:21:38 PM	9/14/99 1:41:15 PM	8	

The Last Merged, Last Modified, and Created columns are updated to reflect any changes occurring as a result of a merge operation. For example, the merging of a single session file into a target file would update the Last Merged column information for the source file (single session), and update the Last Modified column information for the target file to be the current date and time. If the target file had been created during the merge operation, the Created column would be updated to reflect this.

The icons to the left of the filenames are also modified according to any merge operations. The icon for an unmerged single session data file (collected using E-Run) is a single page displaying the E-Prime logo. After an E-Prime single session data file has been merged to another file, the icon is changed to display a green checkmark on the E-Prime logo. The icon for a merged data file is an E-Prime logo on multiple pages. The icon for a file set as the target file receives a bulls-eye on top of the E-Prime logo. Refer to section 4.3.6 for a description of E-Merge icons.

Copying Merge Results

The text in the Merge Log view may be copied and pasted to another application in order to save the detailed results from a merge operation. To copy text to the clipboard, highlight the desired text, and then simultaneously press the Ctrl key and the "C" key. Once copied to the clipboard, the text may be pasted into a text editor.

4.3.7.5 Undo Merge

E-Merge supports one level of undo so that only the most current operation may be undone. After a merge operation is performed, it may be undone until another merge operation is performed or until the application is closed. The most recent merge operation may be reversed (i.e., negated) using the Undo Merge tool button, or the Undo Merge command from the Edit menu.

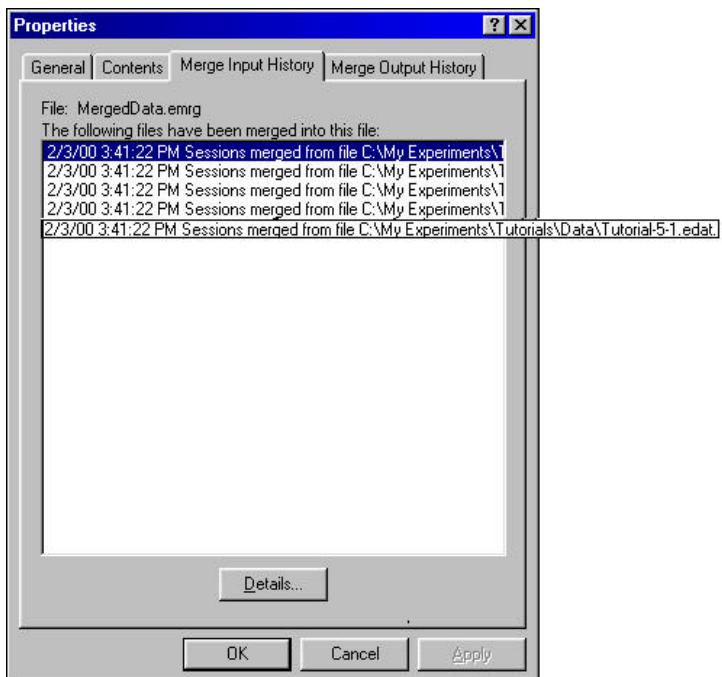


If a merge operation that cannot be undone is attempted, a warning is displayed prior to the completion of the merge operation, and the merge operation may be cancelled.

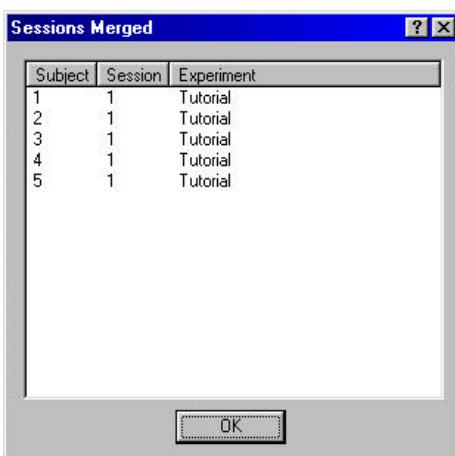


4.3.7.6 Merge Input History

The Merge Input History for a particular file indicates which sessions of data have been merged into that file. To view the Merge Input History for a file, use the Properties tool button on the toolbar or the Properties command under the File menu to display the Properties dialog. In the Properties dialog, select the Merge Input History tab to bring this tab to the front.



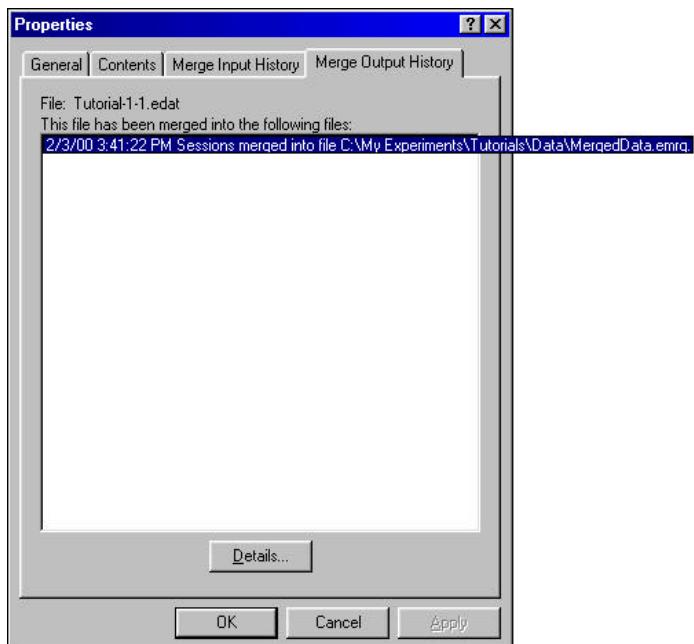
The input history is listed per file rather than per session. Therefore, although a merged file may bring in more than one session of data in the process of being merged to a target file, only the source file (i.e., the merged file) would be listed on the Merge Input History tab. To view the individual sessions of data imported with a merged data file, select the merged data file on the Merge Input History tab and click the Details button. The individual sessions of data included within that merged file (and therefore merged into the target file) will be displayed.



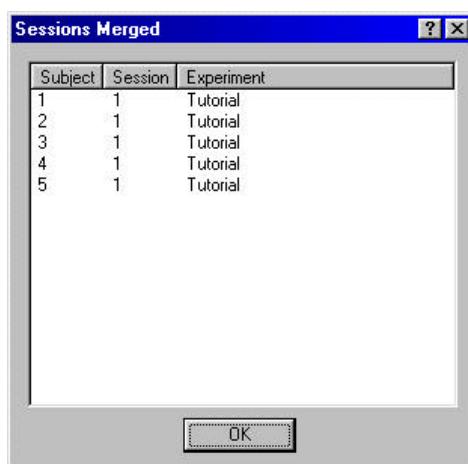


4.3.7.7 Merge Output History

The Merge Output History for a particular file indicates the files to which the selected file has been merged. To view the Merge Output History for a file, select the file in the File List view and use the Properties tool button on the toolbar, or the Properties command under the File menu to display the Properties dialog. In the Properties dialog, select the Merge Output History tab to bring this tab to the front.



The output history is listed per file rather than per session. Therefore, the Merge Output History tab for a merged data file would not list the individual sessions of data contained within that merged data file. To view the individual sessions contained within a merged data file (and therefore output to the target file), select the data file on the Merged Output History tab, and click the Details button. The individual sessions of data included within the merged file will be displayed.





4.3.7.8 Security Levels

The E-DataAid application offers a security feature allowing a user to set security options when supplying a valid password. For example, in E-DataAid, a user may set the security options to disallow edits to subject and session number, or to disallow all edits. When a source file containing a variable with a certain security level is merged into a target file containing the same variable with a different security level, the variable in the target file will have the security level that is the stricter of the two files upon completion of the merge operation.

4.3.8 Moving Around in E-Merge

E-Merge supports basic Windows® methods of navigation using the mouse or the keyboard.

4.3.8.1 Folder Tree View

Keyboard

To activate the Folder Tree view, simultaneously press the **Alt** key and the **0** (zero) key. To collapse a drive or folder, use the left arrow key. To expand a drive or folder, press the right arrow key. Use the up and down arrows to move up or down respectively in the Folder Tree view. Use the **Enter** key to open a selected folder.

Mouse

To expand or collapse a drive, click on the plus next to the drive. To open a folder, click on that folder.

Refresh

The Refresh tool button refreshes the drive of the currently opened file in the Folder Tree view and updates the contents of the File List view.



4.3.8.2 File List View

Keyboard

To activate the File List view, simultaneously press the **Alt** key and the **1** (one) key. Use the up or down keys to select a file. To select a consecutive group of files, hold down the **Shift** key while using the arrow keys. Selecting non-consecutive files via the keyboard is not supported.

Mouse

To select a file, click on the file name. To select a group of consecutive files, click on the first file name with the mouse, hold down the **Shift** key, and click on the last file name with the mouse. All file names between the first and the last file name will be highlighted (selected). To select multiple nonconsecutive files, hold down the **Ctrl** key and click the names of the files to be selected by the mouse. To deselect a file, hold down the **Ctrl** key and click the name of the file to be deselected.



Refresh

The Refresh tool button refreshes the drive of the currently opened file in the Folder Tree view and updates the contents of the File List view.

4.3.8.3 Merge Log View

Keyboard

To activate the Merge Log view, simultaneously press the **Alt** key and the **2** key. Scroll through the text using the up and down arrow keys. Text may be highlighted by holding down the **Shift** key and using the arrow keys to select the desired text. The highlighted text may then be copied to the clipboard by simultaneously pressing the **Ctrl** key and the **C** key. Once copied to the clipboard, the text may be pasted into a text editor.

Mouse

To scroll through the text in the Merge Log view, use the scroll bar on the right side of the View window. To copy text to the clipboard, highlight the text by clicking and dragging across the desired text, and then simultaneously press the **Ctrl** key and the **C** key. Once copied to the clipboard, the text may be pasted into a text editor.

4.3.8.4 Shortcuts

Command	Short Cut Keys
Open	Ctrl + O
Merge	Ctrl + M
Set as Target	Ctrl + T
Properties	Alt + Enter
Undo	Ctrl + Z
Select Unmerged	Ctrl + U
Select All	Ctrl + A
Refresh	F5
Folder Tree	Alt + 0
File List	Alt + 1
Merge Log	Alt + 2
Help Topics	F1

4.3.9 Opening E-Merge

E-Merge may be opened by selecting the E-Merge application from the E-Prime menu accessed via the Start menu, or through the Tools menu in E-Studio.





4.3.10 Opening a Data File

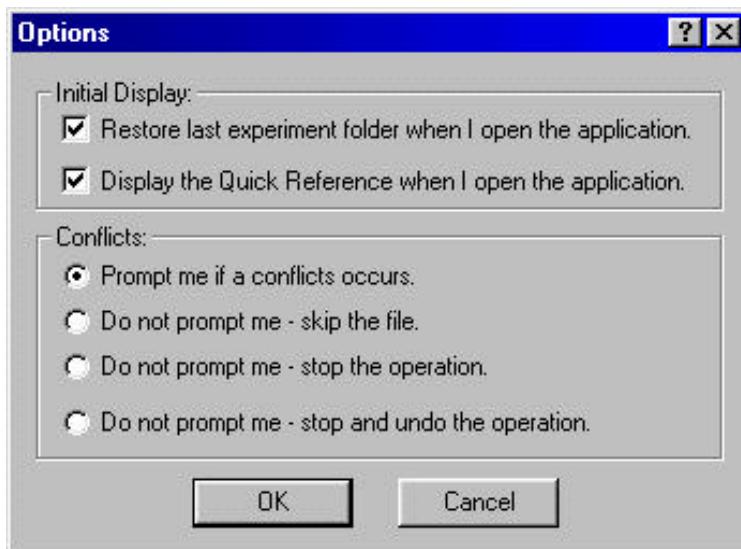
The Open command under the File menu opens the file selected in the File List view into E-DataAid. Only one file may be opened at a time. If no file is selected, more than one file is selected, or an invalid file is selected, the Open command is disabled. Once opened in E-DataAid, modifications may be made to the file, if necessary (e.g., renaming variables, changing subject numbers to avoid conflicts, etc.).

4.3.11 Options

The options for E-Merge may be set using the Options command under the View menu.

4.3.11.1 Initial Display Folder

By default, the E-Merge application opens to display the folder used before the application was last exited. If unchecked, the application will open to the system's current folder.



4.3.11.2 Conflicts

Before merging any data file into another, E-Merge checks for incompatibilities or inconsistencies (i.e., conflicts) between the two files. If E-Merge detects a conflict between a source file and a target file, a course of action must be determined before the merge operation may proceed. By default, if any conflicts arise during a merge operation, the merge process will be interrupted and the user will be prompted to determine a course of action.

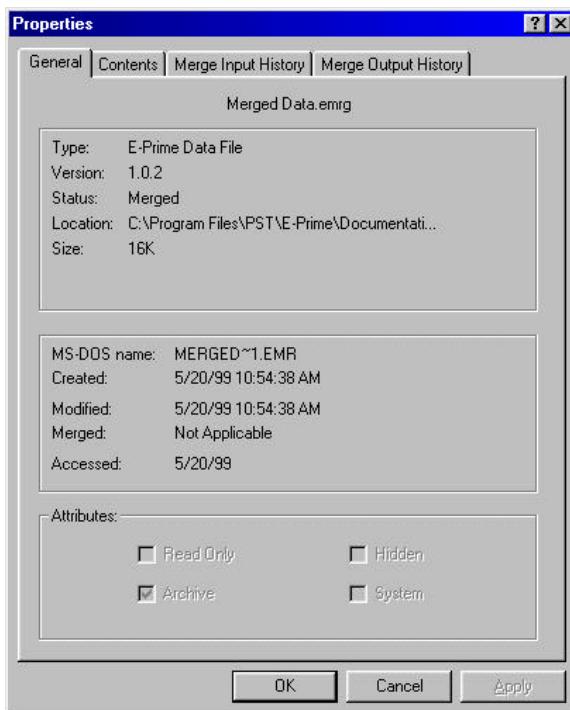
Interruptions for each and every conflict occurring may be avoided by determining a course of action prior to the merge operation. When predetermining the course of action, a single course of action may be chosen which will be applied to any conflict encountered. The course of action for all conflicts occurring may be predetermined to: 1) skip the file causing the conflict and continue with the merge operation, 2) stop the merge operation at the point at which the conflict occurred, or 3) negate the current merge operation. If the merge operation is stopped at the point of conflict, all files merged prior to the occurrence of the conflict remain merged in the target file. If the merge operation is negated, all files merged prior to the conflict are unmerged. In all cases, if E-Merge encounters a conflict during a merge operation, E-Merge will display the cause of the conflict and the course of action taken in the Merge Log view at the end of the merge operation.



4.3.12 Properties

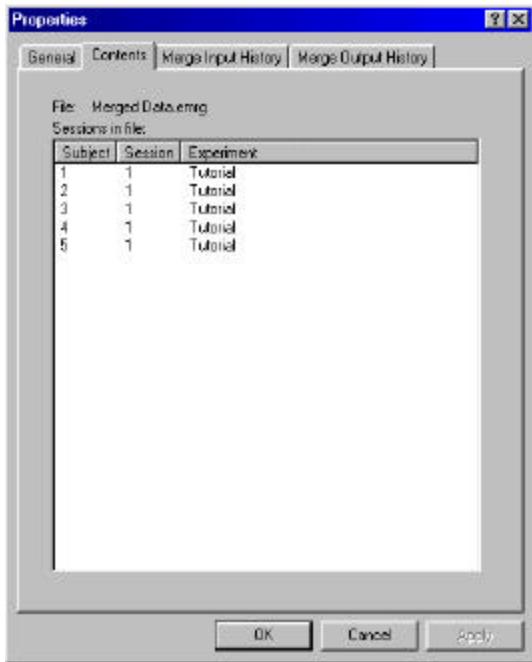
4.3.12.1 General

The General tab displays the filename, size, location, and version of the selected file, whether the file contains a single session or multiple sessions of data (merged data), information concerning the creation date of the file, and the dates the file was last accessed, modified, or merged to another file. The MS-DOS name field displays the name of the file in a format that can be used with MS-DOS applications. The Attributes section displays characteristics of the file, for example, if the file is read-only or archived.



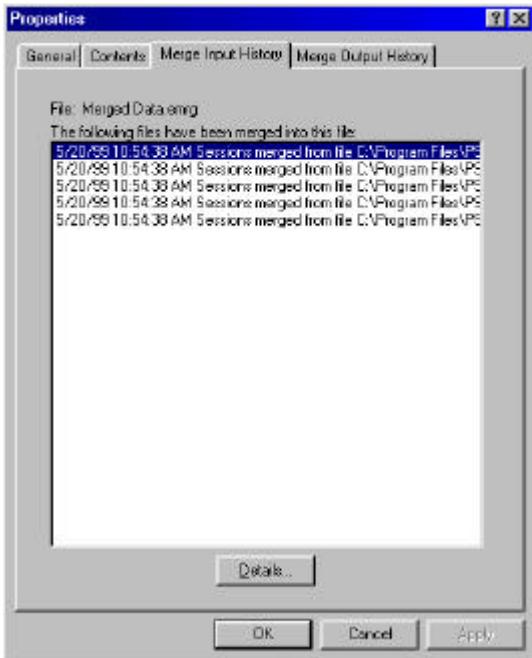
4.3.12.2 Contents

The Contents tab lists the subject, session, and experiment name information for each single session data file contained within the selected file. If the selected file is a single session data file, the Contents tab would display the information for only that file. If the selected file is a merged data file, the Contents tab would display information for all sessions of data contained within that file. The information on the Contents tab may be sorted by column by clicking on the column header.



4.3.12.3 Merge Input History

The Merge Input History tab displays all of the files that have been merged into the selected file in chronological order, including both single session and merged data files. The input history is listed on a file basis rather than by session. Therefore, the input of a merged data file may include the import of more than a single session of data. The information for the individual sessions of data merged through the input of a merged file may be viewed by selecting the merged file on the Merge Input History tab and clicking the Details button.





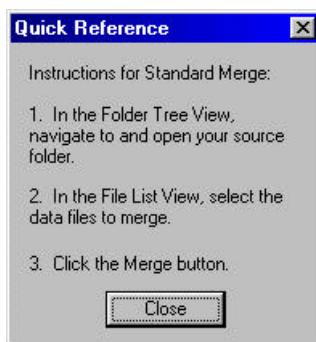
4.3.12.4 Merge Output History

The Merge Output History tab displays the name of any files to which the selected file has been merged. The output history is listed on a file basis rather than by session. Therefore, the merging of a merged data file to another file may include more than a single session of data (i.e., the merged data file may have contained more than one session of data, which would have been merged to the other file). The individual sessions merged to the target file through a merge operation involving a merged data file may be viewed by selecting the merged file on the Merge Output History tab and clicking the Details button.



4.3.13 Quick Reference

The Quick Reference dialog guides the user through the most common type of merge operation (i.e., standard merge), in which all unmerged files in a folder are merged into a master data file for analysis. The Quick Reference dialog is meant to serve as an aid in learning how to use E-Merge, as well as a reminder of the basic steps involved in a merge operation once the application has been mastered.





The Quick Reference dialog appears by default when the E-Merge application is opened. This default may be changed using the Options command under the View menu so that the dialog is not displayed when E-Merge is opened. The dialog may be redisplayed at any time using the Quick Reference command under the View menu, and may be repositioned on the screen by clicking on the title bar and dragging the dialog to a new position.



E-Prime Reference Guide
Chapter 4: E-Merge

This page intentionally left blank.



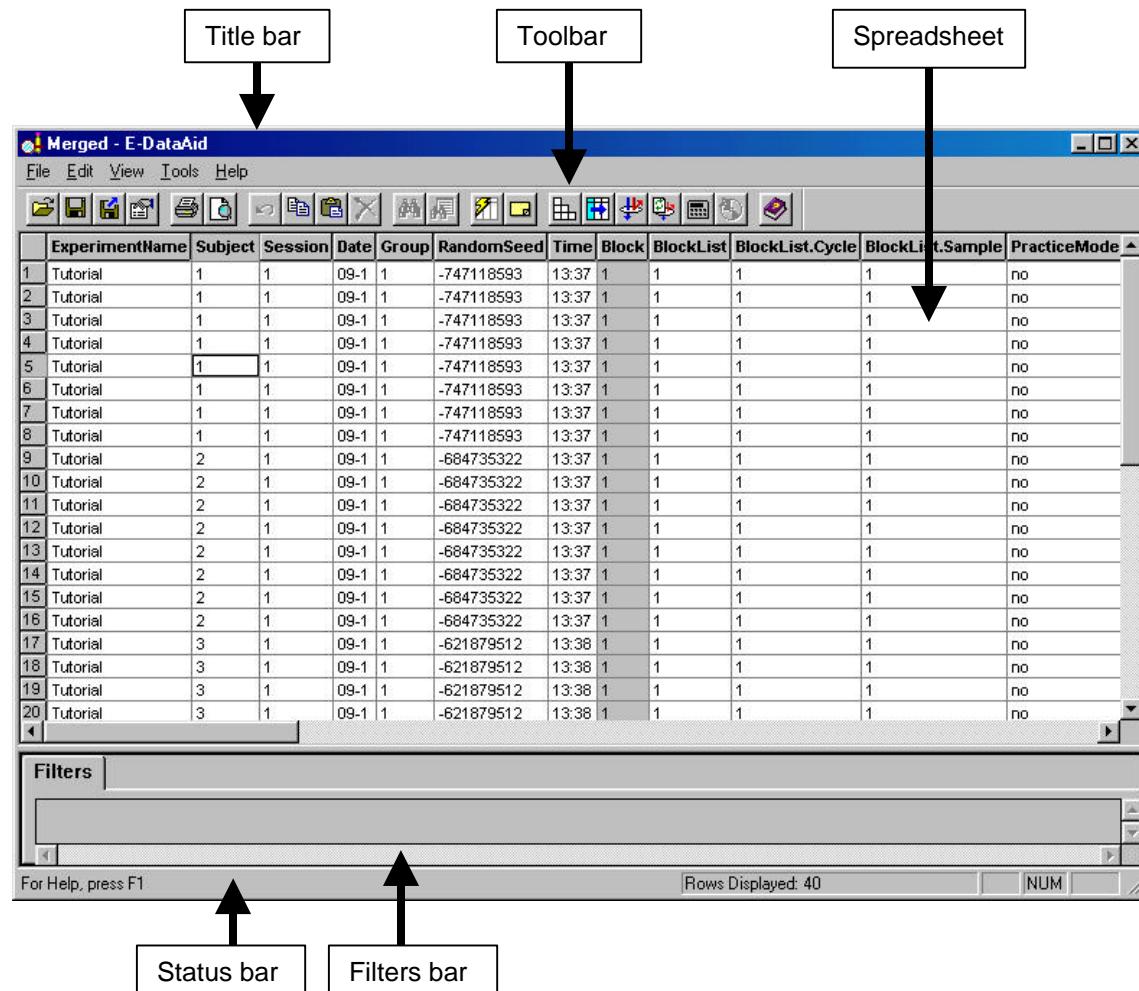
Chapter 5: E-DataAid

5.1 About the E-DataAid Application

The E-DataAid application allows viewing and editing of E-Prime data files. This application is necessary because E-Prime uses a proprietary, binary data file format, which prevents direct viewing or manipulation of the data via a common application such as a text editor.

5.2 About the Interface

The title bar is displayed at the top of the application screen. The title bar displays the E-DataAid icon, the name of the open file, and the name of the application (E-DataAid). The menu bar is displayed at the top of the application screen below the title bar. The menu bar lists the menus available within the E-DataAid application, including the File, Edit, View Tools, and Help menus. Other features include: the toolbar, spreadsheet, Filters bar and status bar.





5.2.1 Menus

5.2.1.1 File Menu

Command	Shortcut	Function
<u>Open...</u>	Ctrl+O	Opens an existing E-Prime data file.
<u>Admin Open...</u>		Opens a file as an Administrator, allowing access to security restrictions on the file.
<u>Close</u>		Closes the currently opened file but does not exit E-DataAid.
<u>Save</u>	Ctrl+S	Saves the opened file to the current name.
<u>Save As...</u>		Displays the Save As dialog, which allows the file to be saved to a specific name or folder.
<u>Import</u>		Imports text files containing raw data collected using MEL Professional, PsyScope, or E-Prime.
<u>Export...</u>		Exports all displayed (unhidden) columns and rows in the E-DataAid spreadsheet to a text file, with flexibility of formatting to accommodate import requirements from other packages.
<u>Print Setup...</u>		Allows the setting of print properties (Printer, Paper, Orientation).
<u>Print Preview</u>		Allows the viewing of the document as it would appear in printed form.
<u>Print...</u>	Ctrl+P	Sends the currently displayed spreadsheet to the printer.
<u>Properties</u>		Displays the properties for the opened file.
<u>File Security...</u>		Allows the setting of file security settings. This command is only active when logging in as an Administrator.
<u>Change Password...</u>		Allows the user to change the password required to log in as an Administrator.
<u>Recent File History</u>		List of the most recently accessed files.
<u>Exit</u>		Exits the application.

5.2.1.2 Edit Menu

Command	Shortcut	Function
<u>Undo</u>	Ctrl+Z	Reverses the most recent edit operation performed. E-DataAid supports undo for all modifications made during the current E-DataAid session (i.e., until the application is closed). When an edit operation is undone, the audit trail corresponding to that operation is also deleted.
<u>Copy</u>	Ctrl+C	Copies the contents of the selected items to the clipboard.
<u>Paste</u>	Ctrl+V	Pastes the contents of the clipboard into the selected area.
<u>Delete</u>	Ctrl+D	Designates the selected cell(s) as having missing data ("NULL"). Available only for cells containing data which may be edited.
<u>Select Column</u>	Ctrl+F	Highlights the column in which the currently selected cell is located.
<u>Find...</u>	Ctrl+F	Displays the Find dialog box in order to search for specific text.
<u>Replace...</u>	Ctrl+H	Searches a range of cells, columns, or rows for specific text to be replaced by a designated string.
<u>Fill Down</u>		Fills all cells in the selected range within a single column with the contents of the first cell in the selected range. Selected range may be only one column wide. Only available for cells containing data which may be edited.
<u>Add Variable...</u>		Displays the Add Variable dialog box in order to enter the name, level, and data type for the creation of a new variable at the end of the spreadsheet.
<u>Rename Variable...</u>		Displays the Rename Variable dialog box in order to enter the new name for a variable. Not available for session level variables (i.e., Subject, Session, or ExperimentName).



5.2.1.3 View Menu

Command	Shortcut	Function
Toolbar		The Toolbar command turns the display of the toolbar on or off depending on its current state. A checkmark next to the Toolbar command indicates the toolbar is displayed.
Status Bar		The Status Bar command turns the display of the status bar on or off depending on its current state. A checkmark next to the Status Bar command indicates the status bar is displayed.
Hide Columns		Hides the currently selected columns.
Unhide Columns		Unhides (i.e., redisperslays) all hidden columns.
Restore Column Order		Restores the columns to the default order: Experiment Name, Subject, Session, and all remaining variables in alphabetical order within their hierarchical level (e.g., session, block, trial).
Restore Spreadsheet		Clears all filters, unhides all columns, and restores the columns to the default order.
Display Annotations...		Displays the Annotations dialog box in order to view and add annotations.
Options...		Displays the Options dialog box in order to view and change the application's display options. The View options allow the user to set options concerning the font and color of the display, whether to prompt the user to enter an annotation when modifications are made, and options concerning the display of data in the spreadsheet.

5.2.1.4 Tools Menu

Command	Shortcut	Function
Collapse Levels...		Displays the Collapse Levels dialog box in order to collapse all levels below a designated level. Collapsing levels automatically restores the columns to the default order, unhides all columns, clears all filters, filters all levels below the designated level on instance 1 or NULL, and hides all columns below the designated level.
Arrange Columns...		Displays the Arrange Columns dialog box in order to unhide, hide, and arrange columns within the spreadsheet display.
Filter...		Displays the Filter dialog box in order to apply new filters to the spreadsheet, and clear active filters. Filters may also be applied or cleared by double clicking the column header, or by right-clicking the column header and selecting the Filter command in the context menu.
Refresh Filters		Refreshes the filters to include edits or changes in the display.
Analyze...		Displays the Analyze dialog box in order to specify and run an analysis. The results of each analysis are displayed in a table format that may be copied to the clipboard or exported to a text file.
Batch Analysis		Displays the Batch Analysis dialog in order to specify and run a group of saved analyses in succession.
Play	F5	Plays the WAV sound file in the selected cell.

5.2.1.5 Help Menu

Command	Shortcut	Function
Help Topics	F1	Displays the E-DataAid Help system.
About E-DataAid...		Displays a dialog box indicating the version information for the application.

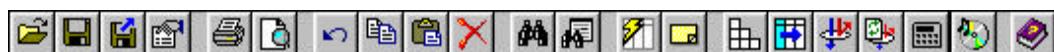


5.2.1.6 Context Menu

A context menu may be enabled by right-clicking on a column heading. The context menu includes the Hide, Filter, Find, Replace, and Rename Variable commands.

5.2.2 Toolbar

The toolbar is displayed at the top of the application display below the title bar and menu bar, and may be toggled on or off using the Toolbar command in the View menu.



The toolbar includes shortcuts to frequently used functions in E-DataAid.

Tool Button	Command	Tool Button	Command
	Open		Replace
	Save		Restore Spreadsheet
	Export		Display Annotations
	Properties		Collapse Levels
	Print		Arrange Columns
	Print Preview		Filter
	Undo		Refresh Filters
	Copy		Analyze
	Paste		Play
	Delete		Help
	Find		

5.2.3 Spreadsheet

The spreadsheet is the main view in the E-DataAid application, and appears in the center of the application display. The spreadsheet displays the data included within the currently opened file in a grid format similar to Excel. The columns in the spreadsheet represent the experiment variables, and the rows represent the lowest log level in the experiment (e.g., trials). When E-DataAid is first opened, the spreadsheet remains gray until a file is opened, at which time the data within the opened file is displayed in the grid format.



ExperimentItem	Subject	Session	Date	Group	RandomSeed	Time	Block	BlockList	BlockList_Cycle	BlockList_Sample	PracticeMode	ProcedureBlock	RunningBlocks	Trial	Co
1	Tutorial	1	09-1	1	-747118993	13:37	1	1	1	1	no	BlockIPrec	BlockList	1	1
2	Tutorial	1	09-1	1	-747118993	13:37	1	1	1	1	no	BlockIPrec	BlockList	2	2
3	Tutorial	1	09-1	1	-747118993	13:37	1	1	1	1	no	BlockIPrec	BlockList	3	1
4	Tutorial	1	09-1	1	-747118993	13:37	1	1	1	1	no	BlockIPrec	BlockList	4	1
5	Tutorial	1	09-1	1	-747118993	13:37	1	1	1	1	no	BlockIPrec	BlockList	5	1
6	Tutorial	1	09-1	1	-747118993	13:37	1	1	1	1	no	BlockIPrec	BlockList	6	1
7	Tutorial	1	09-1	1	-747118993	13:37	1	1	1	1	no	BlockIPrec	BlockList	7	2
8	Tutorial	1	09-1	1	-747118993	13:37	1	1	1	1	no	BlockIPrec	BlockList	8	2
9	Tutorial	1	09-1	1	-747118993	13:37	1	1	1	1	no	BlockIPrec	BlockList	9	2
10	Tutorial	2	09-1	1	-684735322	13:37	1	1	1	1	no	BlockIPrec	BlockList	1	2
11	Tutorial	2	09-1	1	-684735322	13:37	1	1	1	1	no	BlockIPrec	BlockList	2	1
12	Tutorial	2	09-1	1	-684735322	13:37	1	1	1	1	no	BlockIPrec	BlockList	3	1
13	Tutorial	2	09-1	1	-684735322	13:37	1	1	1	1	no	BlockIPrec	BlockList	4	2
14	Tutorial	2	09-1	1	-684735322	13:37	1	1	1	1	no	BlockIPrec	BlockList	5	1
15	Tutorial	2	09-1	1	-684735322	13:37	1	1	1	1	no	BlockIPrec	BlockList	6	2
16	Tutorial	2	09-1	1	-684735322	13:37	1	1	1	1	no	BlockIPrec	BlockList	7	1
17	Tutorial	2	09-1	1	-684735322	13:37	1	1	1	1	no	BlockIPrec	BlockList	8	2
18	Tutorial	3	09-1	1	-621679612	13:38	1	1	1	1	no	BlockIPrec	BlockList	1	1
19	Tutorial	3	09-1	1	-621679612	13:38	1	1	1	1	no	BlockIPrec	BlockList	2	2
20	Tutorial	3	09-1	1	-621679612	13:38	1	1	1	1	no	BlockIPrec	BlockList	3	1
21	Tutorial	3	09-1	1	-621679612	13:38	1	1	1	1	no	BlockIPrec	BlockList	4	1
22	Tutorial	3	09-1	1	-621679612	13:38	1	1	1	1	no	BlockIPrec	BlockList	5	1
23	Tutorial	3	09-1	1	-621679612	13:38	1	1	1	1	no	BlockIPrec	BlockList	6	2
24	Tutorial	3	09-1	1	-621679612	13:38	1	1	1	1	no	BlockIPrec	BlockList	7	2
25	Tutorial	3	09-1	1	-621679612	13:38	1	1	1	1	no	BlockIPrec	BlockList	8	2
26	Tutorial	4	09-1	1	-559869552	13:39	1	1	1	1	no	BlockIPrec	BlockList	1	1
27	Tutorial	4	09-1	1	-559869552	13:39	1	1	1	1	no	BlockIPrec	BlockList	2	1
28	Tutorial	4	09-1	1	-559869552	13:39	1	1	1	1	no	BlockIPrec	BlockList	3	1
29	Tutorial	4	09-1	1	-559869552	13:39	1	1	1	1	no	BlockIPrec	BlockList	4	2
30	Tutorial	4	09-1	1	-559869552	13:39	1	1	1	1	no	BlockIPrec	BlockList	5	1
31	Tutorial	4	09-1	1	-559869552	13:39	1	1	1	1	no	BlockIPrec	BlockList	6	2
32	Tutorial	4	09-1	1	-559869552	13:39	1	1	1	1	no	BlockIPrec	BlockList	7	1
33	Tutorial	5	09-1	1	-498629390	13:40	1	1	1	1	no	BlockIPrec	BlockList	8	2

5.2.3.1 Organization

By default, the spreadsheet displays the ExperimentName, Subject, and Session variables in the first three columns in the grid. The remaining variables are organized alphabetically within hierarchical levels (e.g., Session, Block, Trial, etc.). The spreadsheet may be reorganized by moving, hiding, or unhiding columns. To retain the new organization of the spreadsheet the next time the file is opened, save the file before closing. Otherwise, the default order of the spreadsheet will be restored the next time the file is opened. If a variable exists at more than one level (e.g., block and trial levels), each level will be placed in a separate column, and the column header will display the variable's name followed by its level number or name in brackets (i.e., whether the level appears as a number or a name is determined by how the application options are set).

	Procedure[Block]	Procedure[Trial]
1	BlockProc	TrialProc
2	BlockProc	TrialProc
3	BlockProc	TrialProc
4	BlockProc	TrialProc
5	BlockProc	TrialProc
6	BlockProc	TrialProc
7	BlockProc	TrialProc
8	BlockProc	TrialProc
9	BlockProc	TrialProc
10	BlockProc	TrialProc
11	BlockProc	TrialProc
12	BlockProc	TrialProc
13	BlockProc	TrialProc
14	BlockProc	TrialProc
15	BlockProc	TrialProc
16	BlockProc	TrialProc
17	BlockProc	TrialProc
18	BlockProc	TrialProc
19	BlockProc	TrialProc
20	BlockProc	TrialProc
21	BlockProc	TrialProc
22	BlockProc	TrialProc
23	BlockProc	TrialProc
24	BlockProc	TrialProc



Each instance of the lowest level of data is displayed in its own row (i.e., if the trial level is the lowest level, each row would represent a trial). To maintain the consecutive nature of the data, the application does not allow the movement of rows, but rows may be hidden or displayed by using filters.

5.2.3.2 Cells

Editing

A cell's value may be edited by typing directly in a particular cell. If a cell cannot be edited, it will appear in gray, and the cursor cannot be placed in the cell.

Trial
1
2
3
4
5
6
7
8

Any cell in a column displaying level numbers for any level below the session level will be read-only (e.g., Block Number, Trial Number, etc.). Because of the hierarchical nature of experiment data, the application does not allow editing of these numbers. Any cell in a column on which an Administrator places a security restriction will also be read-only.

Inheritance

If a cell contains italicized text, the application's inheritance option is on, and the value contained within that cell has been inherited from another cell. When the inheritance option is on, any multi-level variable cell that has missing data will inherit its data value from the cell in the same row for that variable's next highest level. For example, if a variable occurs at two levels (e.g., Color occurs at both the Block and Trial levels), the spreadsheet will have two columns for that variable (e.g., Color[Block] and Color[Trial]). When inheritance is on, any cells containing missing values in the lower level for the variable (e.g., Trial) will inherit the value from the next higher level (e.g., Block). Inherited values are displayed in italics.

Color[Block]	Color[Trial]
blue	<i>blue</i>
red	<i>red</i>

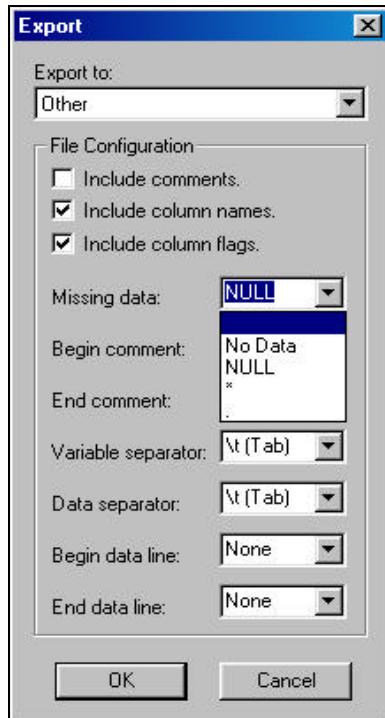


When inheritance is off, the cells would contain NULL. The inheritance option may be toggled on or off using the Options command in the View menu.

Color[Block]	Color[Trial]
blue	NULL
red	NULL

Missing Data

Within E-DataAid, any cell containing missing data is displayed as NULL. When exporting data, the Export dialog includes a Missing Data field to allow changing the way missing data is represented, depending on the constraints of the statistical package being used. The options within the Export dialog allow the user to export cells containing missing data as blank cells, cells containing "No Data", "Null", an asterisk (*), a period (.), or a user-entered value.



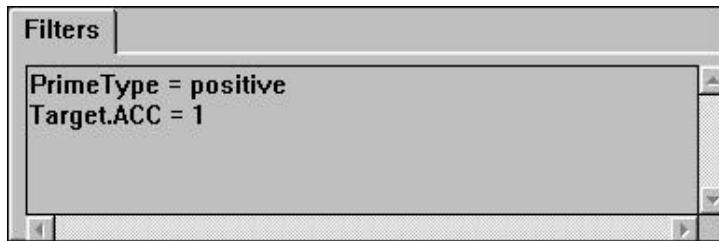
Navigation

To navigate to a particular cell in the spreadsheet, use the arrow keys. To place the cursor in a cell, either click inside the cell using the mouse, or navigate to the cell with the arrow keys and press the F2 key.



5.2.4 Filters Bar

The Filters bar is located at the bottom of the application display below the spreadsheet.



The Filters bar displays any filters currently applied to the data. If no filters are being applied, the Filters bar area remains blank. The Filters bar may be resized by placing the cursor over the border between the Filters bar and the spreadsheet. After the cursor changes to a double pointed arrow (pointing up and down), click and drag the border up or down until the Filters bar is the desired size, and release the mouse button.

The Filters bar area is read-only (i.e., may not be edited). However, the contents of the Filters bar may be copied to the clipboard and pasted into another application.

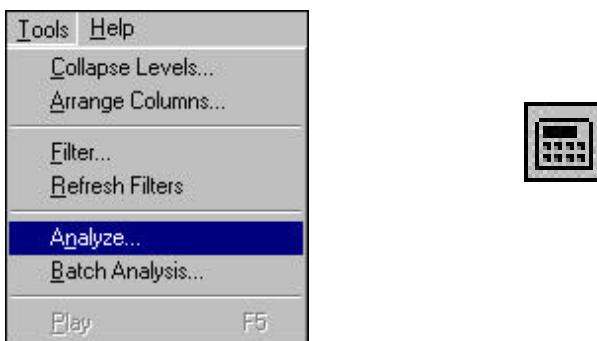
5.2.5 Status Bar

The status bar is located at the bottom of the application display below the Filters bar, and may be toggled on or off using the Status Bar command in the View menu. The status bar displays the state of the application and the number of unhidden rows displayed in the spreadsheet.

5.3 Using E-DataAid

5.3.1 Analyzing

The Analyze command displays the Analyze dialog box, which is used to create and run descriptive statistics using the currently opened spreadsheet. The results of each analysis are displayed in the Table dialog in a table format that may be copied to the clipboard or exported to a text file. The Analyze dialog may be displayed using the Analyze command in the Tools menu, or by clicking the Analyze tool button.



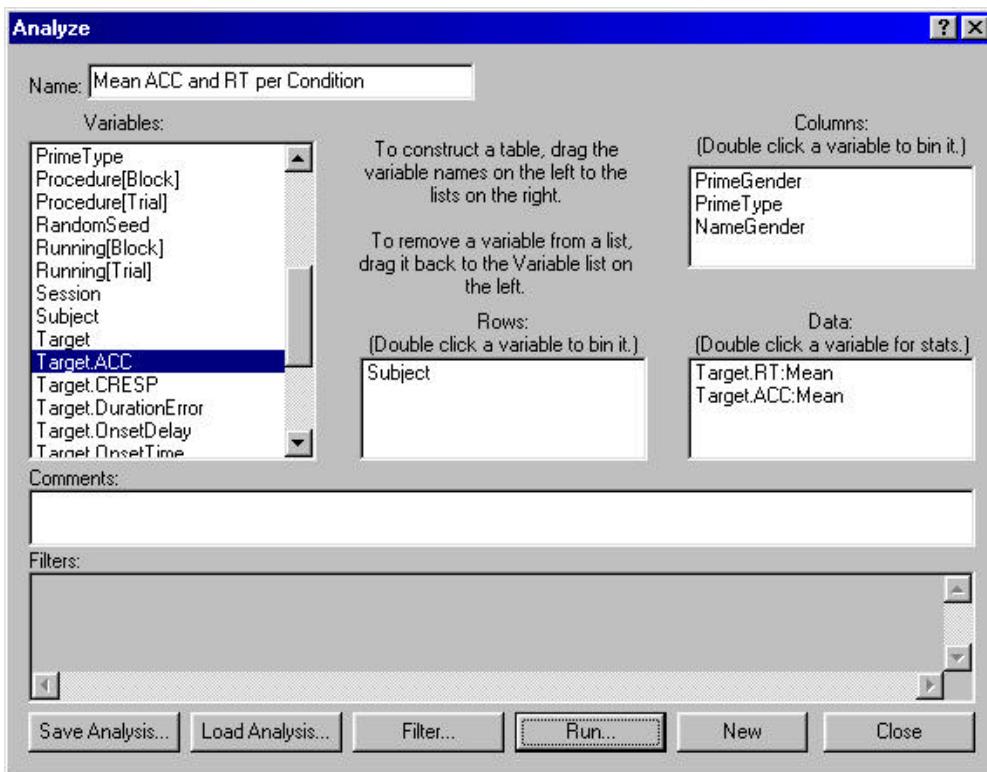
All statistics are calculated based on the filtered spreadsheet. Thus, only displayed data is included in the analysis. If a value cannot be calculated for an analysis, the table of results will



display "N/A." A value may not be able to be calculated if the variable contains all string data, or does not include a sufficient number of observations. If an observation did not occur in the analysis, the table will display a blank value.

5.3.1.1 Creating an Analysis

Within the Analyze dialog, use the mouse or the Tab key to navigate between the available buttons and fields.



Name

Enter a name for the analysis in the **Name** field. The name supplied will be offered as a default name when saving the analysis as an ANL (E-Prime analysis) file. This field is optional, and does not affect the analysis if left blank. Analysis names may be up to 30 characters long, including spaces.

Variables List

The **Variables** list displays all variable names in the spreadsheet. Click and drag variables from the Variables list to enter them in the appropriate field (Column, Row or Data, respectively) in order to achieve the desired analysis. A variable may not have both a row and a column orientation, nor can it appear in the Rows list or Columns list twice. However, a variable may be moved between the Row and Columns lists by dragging a variable from one list and dropping it on the other. To remove a variable from a field, drag the variable from the current field (Column, Row or Data) back to the Variables list.



Columns

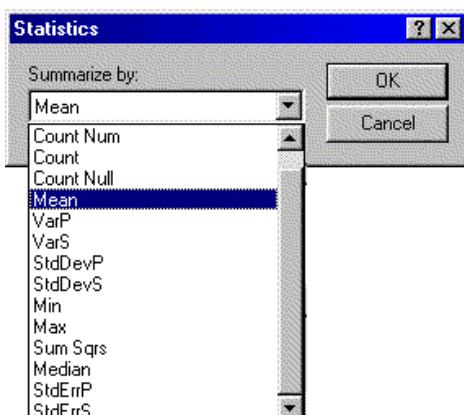
The **Columns** list displays the conditional variables to be used in determining the columns of the analysis. Variables may be reordered within the Columns list by dragging variables from one position and dropping them to their new positions within the list. Use of the Columns list is optional. A separate column will be created for each combination of the levels of every variable in the list. For example, if Variable A (with two levels) and Variable B (with two levels) are added to the Columns list, the analysis will create a column for each combination of the levels of Variable A and Variable B (i.e., A1B1, A1B2, A2B1, A2B2). A variable entered in the Columns list may be binned if the variable contains integer data (see section 5.3.1.2 for Binning Data).

Rows

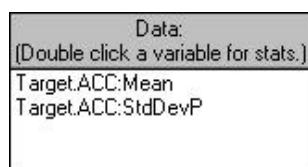
The **Rows** list displays the variables used as the “BY” factor in the experiment. Variables may be reordered within the Rows list by dragging variables from one position and dropping them to their new positions within the list. Use of the Rows list is optional. A separate row will be created for every combination of the levels of the variables in the Rows list. For example, if Variable A (with one level) and Variable B (with three levels) are added to the Rows list, the analysis will create three rows to accommodate each combination of the levels of Variables A and B (i.e., A1B1, A1B2, A1B3). Variables entered in the Rows list may be binned if the variable contains integer data (see section 5.3.1.2 for Binning Data).

Data

The **Data** list displays the dependent measure in the analysis. At least one variable must be entered in the Data list for each analysis. The data for any variables entered in the Data field may be summarized using various statistical measures. Double click any variable in the Data field to designate a method by which that variable’s data may be summarized. By default the data is summarized by the condition mean.



To collect more than one statistic for the same variable, add the variable to the list one time for each statistic desired, and choose a different summary method for each instance of the variable from the Statistics dialog.





The following table lists the statistical measures by which data may be summarized:

Option	Summary value
Count	Number of times a value occurs for the variable whether numeric or string (provided the string is not the string for missing data).
CountNull	Number of times the value for the variable is equal to the string for missing data (NULL).
CountNum	Number of times a numeric value occurs for the variable.
Max	Maximum value
Mean	Mean value
Median	Median value
Min	Minimum value
StdDevP	Population standard deviation
StdDevS	Sample standard deviation
StdErrP	Population standard error
StdErrS	Sample standard error
SumSqrS	Sum of squares
Total	Sum of the individual values for the variable
VarP	Population variance
VarS	Sample variance

Variables may be removed from the Data list by dragging them from the list and dropping them on to the Variables list. Variables cannot be moved from the Data list to the Rows or Columns lists, or vice versa. However, variables may appear in either the Rows or the Columns list in addition to the Data list.

Comments

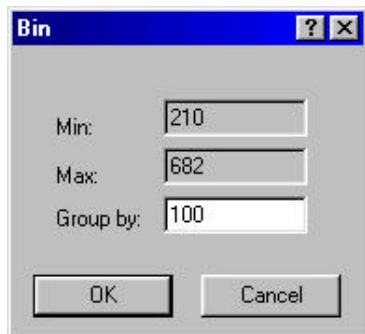
The **Comments** field allows the user to enter up to two lines of additional comments concerning the analysis. Comments are included when an analysis is saved to an ANL file (E-Prime analysis file).

Filters

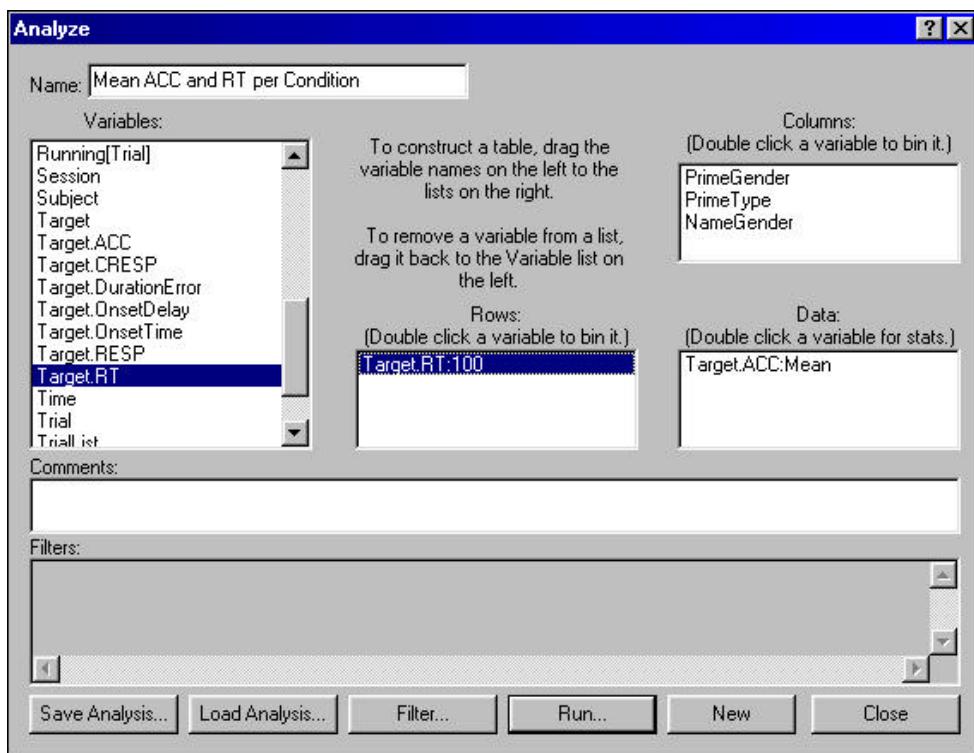
The **Filters** field displays any active filters applied to the spreadsheet. If no filters have been applied to the data, the Filters field remains blank. The Filters field is read-only. In order to apply or remove filters, it is necessary to use the Filter button, or return to the spreadsheet and use the Filter command.

5.3.1.2 Binning Data

A variable placed in the Rows or Columns lists may be binned, providing that the variable contains only integer data. Binning refers to grouping the individual values within a specific variable. For example, a variable containing reaction times might be binned in order to analyze reaction times in groups of 100 milliseconds. Double clicking the variable in the Rows list or Columns list displays the Bin dialog.



The Bin dialog displays the minimum and maximum values for the variable, and contains an editable field (“Group by” field) in which the user may specify a positive integer (1-32762) by which to group the variable’s values. By default, the “Group by” field is set to 1, indicating no binning. To designate a binning value, enter the value in the “Group by” field, and click OK in the Bin dialog. After exiting the Bin dialog, the binned variable (in the Rows or Columns list) will display the name of the variable with the binning value after it.



5.3.1.3 Running an Analysis

The Run button performs the analysis specified in the Analyze dialog. The results of the analysis are displayed in table format in the Table dialog. The results of any analysis may be copied to the clipboard using the Clipboard button, or exported to a text file using the Export button on the Table dialog (See Export).



	A	B	C	D	E	F	G	H
1			female	female	female	female	male	male
2			negative	negative	positive	positive	negative	negative
3	Subject	Stats	female	male	female	male	female	male
4	1	Mean Target RT	386.00	452.00	814.00	1420.00	462.00	385.00
5	2	Mean Target RT	468.00	382.00	901.00	941.00	579.00	304.00
6	3	Mean Target RT	985.00	539.00	359.00	1059.00	994.00	950.00
7	4	Mean Target RT	401.00	1040.00	1060.00	587.00	656.00	509.00
8	5	Mean Target RT	861.00	632.00	667.00	637.00	1178.00	834.00

Conditions

When a table is generated, the Conditions field at the top of the Table dialog indicates the names of the variables with a column orientation. The Conditions field is read-only. In order to change the columns within the table, return to the Analyze dialog and modify the Columns list.

Table

The table portion of the Table dialog contains the results of the analysis. The table is color-coded to aid in the reading of the table.

Display Mode

The Table dialog allows the display to be altered depending on its intended use. The Display Mode options to the right of the table offer three display modes (Plot, StatView and SPSS, and Custom).

Plot – The Plot display mode formats the table so it can be plotted in Excel. This mode is disabled for tables that collect more than one statistic. For example, a table can plot mean reaction times or mean accuracy, but cannot plot both.

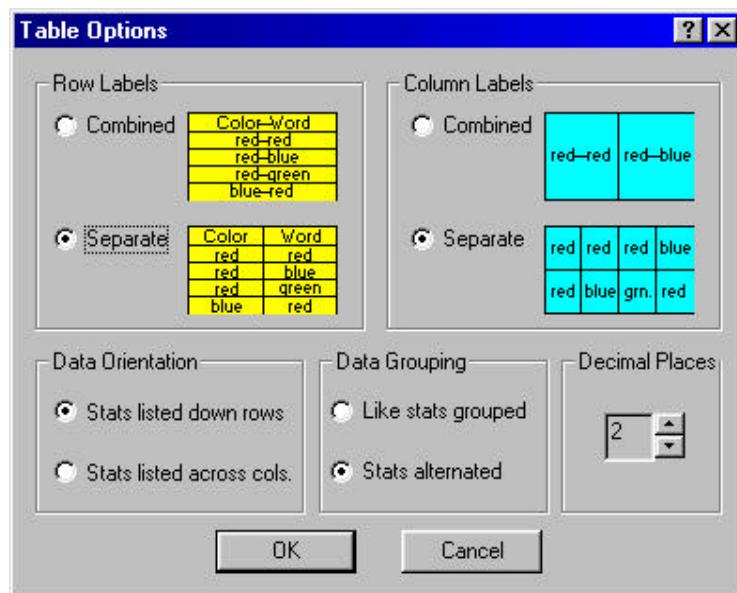
StatView and SPSS – The StatView and SPSS display mode formats the table so it can be exported to a text file for later import into StatView® or SPSS®. For example, a table of Mean RT by Subject may be imported into StatView in order to perform an ANOVA on the table data.

Custom – The Custom display mode formats the table according to user specifications. This is the default setting. Use the Table Options button on the Table dialog to customize the table's format.



Table Options

The Table Options button allows the setting of display properties for a table generated from the Analyze dialog.



Row Labels

Row labels refer to the labels of the columns that contain data collected for the variable(s) in the Row field in the Analyze dialog. These columns are always yellow in the table. For example, if Subject is designated as the only variable in the Row field in the Analyze dialog, then the resulting table would contain at least one yellow column with the label "Subject". If two variables are designated, such as "Color" and "Hand", in the Row field in the Analyze dialog, then the resulting table would contain at least two yellow columns – one with the label "Color" and one with the label "Hand".

Color	Hand
Red	Left
Red	Right
Green	Left
Green	Right

This format is referred to as "Separate Row Labels". However, there may be situations in which only one column will contain the data from all the variable(s) in the Row field in the Analyze dialog, as in the following:

Color-Hand
Red-Left
Red-Right
Green-Left
Green-Right



This format is referred to as “Combined Row Labels”. The columns containing data from all the variable(s) in the Row field in the Analyze dialog are concatenated together into one column.

In the Row Label box on the Table Options dialog, select the “Combined” option for “Combined Row Labels” or select the Separate option for “Separate Row Labels.” This option can only be changed if the Custom display mode is selected for the table. The other two display modes set this option. The default for Custom display mode is “Separate.”

Column Labels

Column labels refer to the labels of the columns that contain data collected for the variable(s) in the Columns Field in the Analyze dialog. These columns are always in blue in the table. Since the table contains one column for every combination of levels for each variable(s) in the Columns Field in the Analyze dialog, the labels for these columns are not the name of the variable(s) but rather the name of the combinations. For example, assume that the data file contains the variable “Color” that has two levels in the data file (i.e., Red and Green). If “Color” is designated as the only variable in the Columns Field in the Analyze dialog, then the resulting table would contain at least two blue columns with the labels “Red” and “Green” respectively appearing in the first row of the table. Assume that the data file also contains the variable “Hand”, and it has two levels in the data file (i.e., Left and Right). If “Color” and “Hand” are designated as variables in the Columns Field in the Analyze dialog, the resulting table would contain at least four blue columns (i.e., one column for each Color/Hand combination “Red/Left”, “Red/Right”, “Gr and “Green/Right”). The “Color” half of the combination would appear in the first row of the table, and the “Hand” half of the combination would appear in the second row of the table.

Red	Red	Green	Green
Left	Right	Left	Right

This format is referred to as “Separate Column Labels”. However, there may be situations in which only one row will contain the combinations from all the variable(s) in the Columns Field in the Analyze dialog.

Red-Left	Red-Right	Green-Left	Green-Right
----------	-----------	------------	-------------

This format is referred to as “Combined Column Labels”. The rows containing combinations from all the variable(s) in the Columns Field in the Analyze dialog are concatenated together into one row.

In the Column Label box on the Table Options dialog, select the Combined option for “Combined Column Labels” or select the Separate option for “Separate Column Labels”. This option may only be changed if the Custom display mode is selected for the table. The other two display modes set this option automatically. The default for Custom display mode is “Separate.”

Data Orientation

Data Orientation refers to the orientation of the labels for the dependent measure(s) in the table. If these labels are displayed with a row orientation (Figure 1), the table treats them like row labels (i.e., they are displayed down rows in yellow). If these labels are displayed with a column orientation (Figure 2), the table treats them like column labels (i.e., they are displayed across columns in blue).



Color	Hand	Stats
Red	Left	Mean RT
Red	Right	Mean RT
Green	Left	Mean RT
Green	Right	Mean RT

Figure 1. Row orientation

Red	Red	Green	Green
Left	Right	Left	Right
Mean RT	Mean RT	Mean RT	Mean RT

Figure 2. Column Orientation

In the Data Orientation box on the Table Options dialog, select the “Stats listed down rows” option to give the labels for the dependent measure(s) a row orientation or select the “Stats listed across columns” option to give these labels a column orientation. This option may only be changed if the Custom display mode is selected for the table. The other two display modes set this option automatically. The default for Custom display mode is “Stats listed down rows.”

Data Grouping

Data Grouping refers to whether the data in the table is grouped or alternated by dependent measure(s). For example, in a table of mean reaction times and mean accuracy rates by subject, all mean reaction times can be grouped together and all mean accuracy rates can be grouped together, or mean reaction times and mean accuracy rates can be alternated by subject.

Color	Hand	Stats
Red	Left	Mean ACC
Red	Left	Mean RT
Red	Right	Mean ACC
Red	Right	Mean RT
Green	Left	Mean ACC
Green	Left	Mean RT
Green	Right	Mean ACC
Green	Right	Mean RT

Figure 1: Alternated Stats

Stats	Color	Hand
Mean ACC	Red	Left
Mean ACC	Red	Right
Mean ACC	Green	Left
Mean ACC	Green	Right
Mean RT	Red	Left
Mean RT	Red	Right
Mean RT	Green	Left
Mean RT	Green	Right

Figure 2: Grouped Stats

In the Data Grouping box on the Table Options dialog, select the “Lik alternated” option. This option may only be changed if the Custom display mode is selected for the table. The other two display modes set this option automatically. The default for Custom display mode is “Stats alternated.”

Please note that the labels for the dependent measure(s) in the table are always arranged alphabetically within the selected grouping regardless of their order in the Data field in the Analyze dialog.

Decimal Places

The Decimal Places option allows the user to change the number of decimal places (0-6) displayed for the statistics. Exceptions are the Count, CountNum and CountNull statistics, which are always displayed as integers. In the Decimal Places box on the Table Options dialog, use the arrows to set the number of decimal places. The default is two decimal places. This option is available for all display modes.



Plot Options

The Plot Options button allows the setting of plot properties for a table generated from the Analyze dialog. This button is enabled only when the table is in Plot display mode. Only tables that contain one dependent measure, such as mean reaction times or mean accuracy, can be plotted.

Type

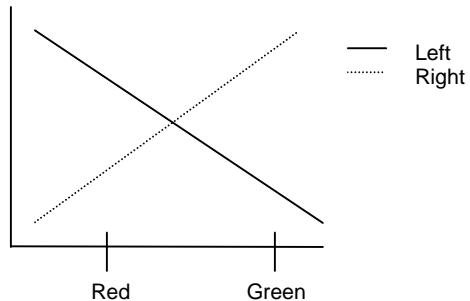
The Type option allows plotting the table as a Column or Line chart. In the Type box on the Plot Options dialog, select the plot type in the drop-down list. The default Type is Column.

Orientation

The Orientation option allows deciding whether the data is plotted by columns or by rows. When plotting by columns (Figure 1), the level combinations of the variable(s) in the Column field of the Analyze dialog make up the chart's legend (i.e., labels from the blue columns of the table). When data is plotted by rows (Figure 2), the level combinations of the variable(s) in the Row field of the Analyze dialog make up the chart's legend (i.e., labels from the yellow columns of the table).

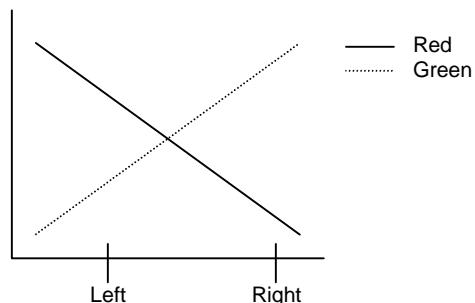
	Mean RT	Mean RT
Color	Left	Right
Red	2000	1000
Green	1000	2000

Figure 1: Column orientation



	Mean RT	Mean RT
Color	Left	Right
Red	2000	1000
Green	1000	2000

Figure 2: Row orientation



In the Orientation box on the Plot Options dialog, select "Column" for a column orientation or "Row" for a row orientation. The default Orientation is Column.

Y-Axis Scale

The Y-Axis Scale option enables or disables the automatic setting of the y-axis scale by the application. By default, this option is set to "Auto", indicating that the application will set the scale. To set the y-axis scale manually, uncheck the box, and enter a minimum and maximum value for the y-axis.



Chart Title

The Chart Title option enables or disables the automatic setting of the title for the plot by the application. By default, this option is set to "Auto", indicating that the application will set the title. To set the title manually, uncheck the box, and enter a chart title.

Y-Axis Title

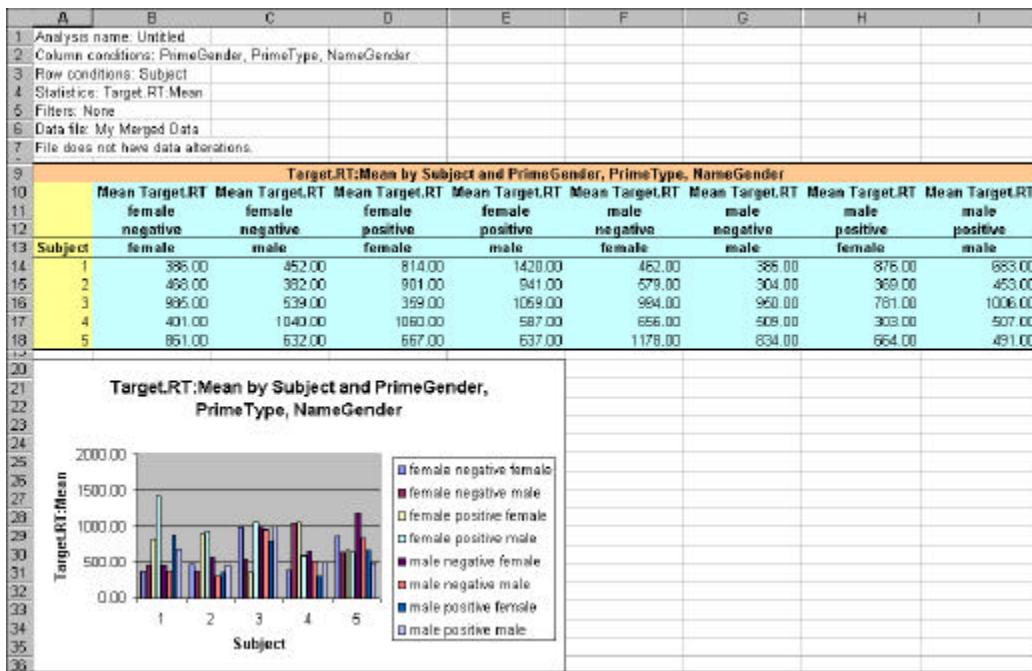
The Y-Axis Title option enables or disables the automatic setting of the title for the y-axis by the application. By default, this option is set to "Auto", indicating that the application will set the title. To set the y-axis title manually, uncheck the box, and enter a title for the y-axis.

Plotting to Excel

Note: This function requires the installation of Excel97 or Excel2000.

The Excel Plot button automatically plots the table in Excel. The Excel Plot button is enabled only when the table is in Plot display mode, and only tables that contain one dependent measure (e.g., mean RT, mean accuracy, etc.) can be plotted.

When plotting in Excel, E-DataAid adds a worksheet to the active workbook. If necessary, E-DataAid activates Excel and/or opens a new workbook. E-DataAid then copies the table to the Excel worksheet, applies formatting, and graphs the data.



Copying to Excel

Note: This function requires the installation of Excel97 or Excel2000.

The Excel Copy button automatically copies the table into Excel. All tables, regardless of the number of dependent measures or display mode, can be copied to Excel. When copying to Excel, E-DataAid adds a worksheet to the active workbook in Excel. If necessary, E-DataAid



activates Excel and/or opens a new workbook. Then, E-DataAid copies the table to the Excel worksheet and formats it.

	A	B	C	D	E	F	G	H	I	J
1	Analysis name: Untitled									
2	Column conditions: NameGender, PrimeGender, PrimeType									
3	Row conditions: Subject									
4	Statistics: Target.RT:Mean									
5	Filters: None									
6	Data file: My Merged Data									
7	File does not have data alterations.									
8										
9	Target.RT:Mean by Subject and NameGender, PrimeGender, PrimeType									
10		female	female	female	female	male	male	male	male	
11		female	female	male	male	female	female	male	male	
12	Subject	Stats	negative	positive	negative	positive	negative	positive	negative	positive
13	1	Mean Target.RT	386.00	814.00	462.00	876.00	452.00	1420.00	385.00	683.00
14	2	Mean Target.RT	468.00	901.00	579.00	369.00	382.00	941.00	304.00	453.00
15	3	Mean Target.RT	985.00	359.00	994.00	781.00	539.00	1059.00	950.00	1006.00
16	4	Mean Target.RT	401.00	1060.00	656.00	303.00	1040.00	587.00	509.00	507.00
17	5	Mean Target.RT	861.00	667.00	1178.00	664.00	632.00	637.00	834.00	491.00

Copying to the Clipboard

The Clipboard button copies the table to the clipboard. The table can then be pasted from the Clipboard into another application.

Exporting the Table

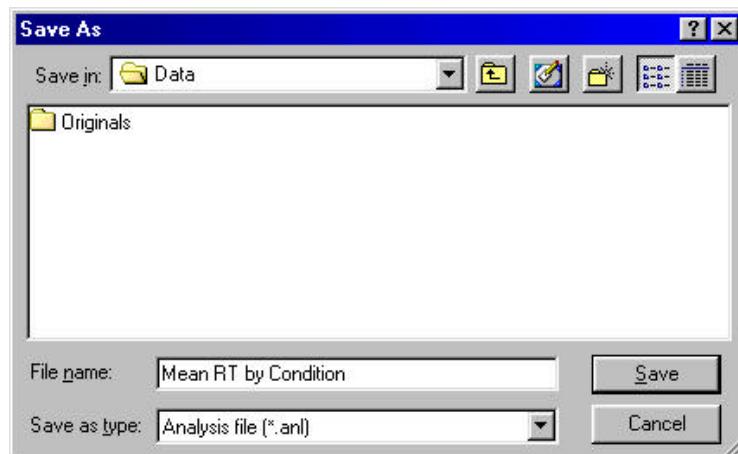
The Export button displays the Export dialog in order to export the data to a text file (see also Export command).

Closing the Table

The Close button closes the Table dialog and returns to the Analyze dialog.

5.3.1.4 Saving an Analysis

The Save Analysis button on the Analyze dialog displays the Save As dialog, with which the specifications for a particular analysis may be saved to a file with the ANL extension.

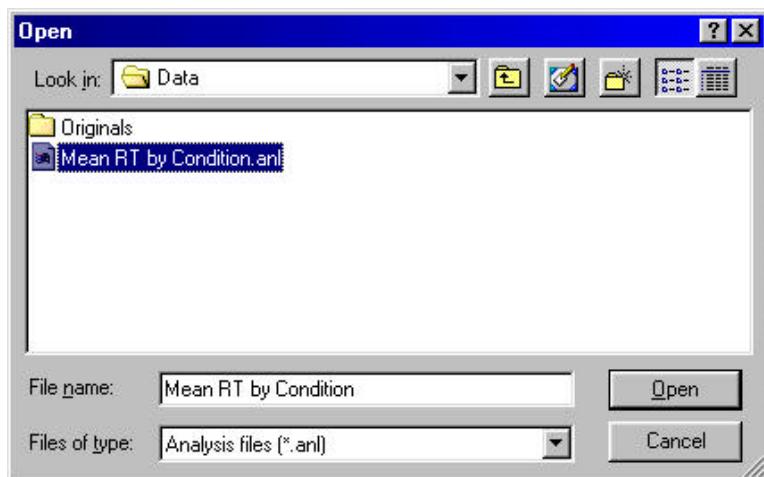




In the Save As dialog, navigate to the folder in which the ANL file will be saved, and enter a name under which to save the analysis. To save the analysis, click the Save button. The application automatically appends the ANL extension to all analysis files. ANL files may later be loaded using the Load Analysis button.

5.3.1.5 Loading an Analysis

The Load Analysis button on the Analyze dialog displays the Open dialog, which allows the specifications for a particular analysis to be loaded from a previously saved ANL file.



To load a specific analysis, navigate to the folder containing the ANL file in which the analysis is saved, select the desired ANL file in the Open dialog, and click **Open**. The E-DataAid application will configure the Analyze dialog with the information saved in the selected ANL file, and will apply any appropriate filters.

5.3.1.6 Creating a New Analysis

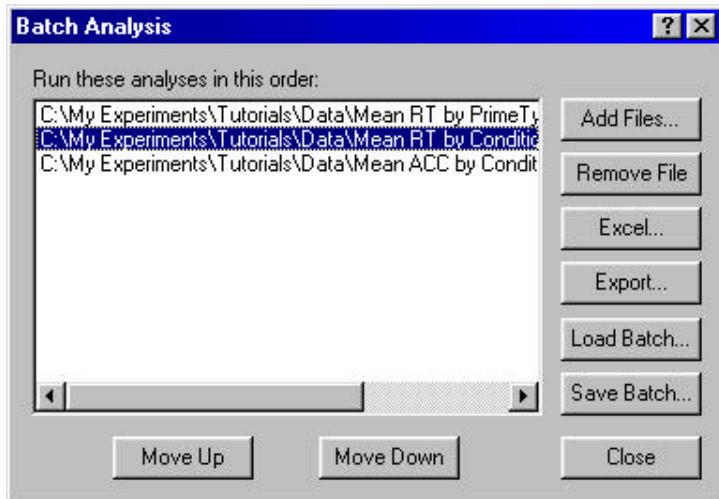
The New button on the Analyze dialog removes any variables from the Row, Column, and Data lists, clears the Name and Comments fields, and clears table and plot options. Filters are not cleared, as filters apply to the spreadsheet and are merely displayed on the Analyze dialog. To clear any filters for a particular analysis, use the Filter command via the Tools menu or the Filter tool button. Before clearing the fields for a new analysis, the user is prompted to save the current settings.

5.3.1.7 Closing the Analyze dialog

The Close button dismisses the Analyze dialog. Before closing the dialog, the user is prompted to save the current settings.

5.3.1.8 Batch Analyses

The Batch Analysis command on the Tools menu displays the Batch Analysis dialog. This dialog allows running a batch of saved analyses in succession. Analyses are created via the Analyze command on the Tools menu. They are saved as ANL files using the Save Analysis button on the Analyze dialog. When running a batch of analyses, the results may either be plotted/copied to an Excel workbook, or output to a text file.



Adding Analyses

Use the Add Files button to add analysis files to the list via a multi-selection Open dialog. The entire list of analyses will be performed when an action is specified.

Removing Analyses

Use the Remove File button to remove a highlighted file from the list. This button is disabled if no files are highlighted.

Reordering Analyses

Use the Move Up and Move Down buttons to reorder a highlighted file in the list. This button is disabled if no files are highlighted.

Copying/Plotting to Excel

Use the Excel button to load each analysis in the list, run it, and copy or plot the results in an Excel workbook. If the analysis was saved with Plot mode, it is plotted. If the analysis was saved with any other display mode, it is copied. This button is disabled if there are no files in the analysis list.

Exporting Analyses

Use the Export button to load each analysis in the list, run it, and dump the results to a tab-delimited file. E-DataAid prompts the user to name the text file to be used for output.

Loading Batch Analyses

Use the Load Batch button to load the names of the analysis files from a batch file created with the Save Batch button. The batch file is a text file with the BAT extension. It contains the names of the analyses to be run.

Saving Batch Analyses

Use the Save Batch button to save the names of the analysis files in the list to a batch file. This batch file can later be read with the Load Batch button. This button is disabled if no files are in the analysis list.

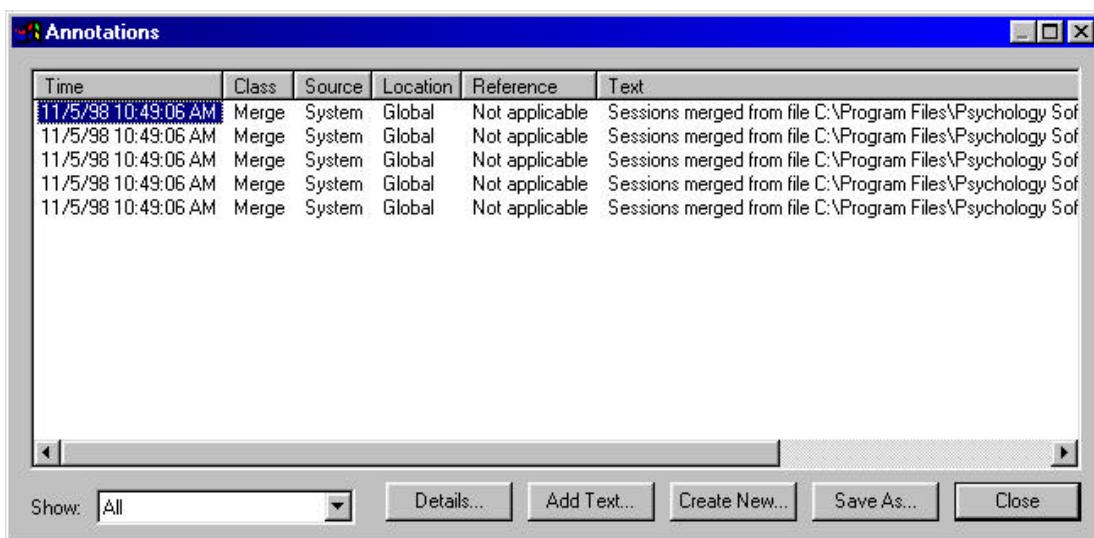


5.3.2 Annotations

Annotations describe modifications made to or processes performed on an E-Prime data file. The Annotations feature is like an electronic version of the experimenter's notebook. When an individual value is modified, an annotation is stored with the session of data containing the modified value, and the modified date for the session is updated. All other annotations are stored globally with the data file. The application does not permit deletion of any annotations except through the Undo command under the Edit menu (i.e., to undo the operation resulting in the creation of an annotation).

5.3.2.1 Displaying Annotations

The Annotations dialog may be displayed by using the Display Annotations command under the View menu or the Display Annotations tool button on the toolbar. The Annotations dialog contains a list of all of the annotations (session and global) within the data file, and displays specific information concerning the annotation.



The screenshot shows the 'Annotations' dialog box. The title bar reads 'Annotations'. The main area is a table with columns: Time, Class, Source, Location, Reference, and Text. There are five rows, each representing an annotation. The 'Time' column shows the date and time as '11/5/98 10:49:06 AM'. The 'Class' column shows 'Merge'. The 'Source' column shows 'System'. The 'Location' column shows 'Global'. The 'Reference' column shows 'Not applicable'. The 'Text' column shows 'Sessions merged from file C:\Program Files\Psychology Sof'. At the bottom of the dialog are buttons for 'Show' (with dropdown menu 'All'), 'Details...', 'Add Text...', 'Create New...', 'Save As...', and 'Close'.

Time	Class	Source	Location	Reference	Text
11/5/98 10:49:06 AM	Merge	System	Global	Not applicable	Sessions merged from file C:\Program Files\Psychology Sof
11/5/98 10:49:06 AM	Merge	System	Global	Not applicable	Sessions merged from file C:\Program Files\Psychology Sof
11/5/98 10:49:06 AM	Merge	System	Global	Not applicable	Sessions merged from file C:\Program Files\Psychology Sof
11/5/98 10:49:06 AM	Merge	System	Global	Not applicable	Sessions merged from file C:\Program Files\Psychology Sof
11/5/98 10:49:06 AM	Merge	System	Global	Not applicable	Sessions merged from file C:\Program Files\Psychology Sof

By default, the annotations are listed chronologically. Data within the Annotations dialog may be sorted by column. Click on the column header to sort the information in ascending or descending order.

Time

The Time column displays the time and date that the application created the annotation.

Class

The Class column displays the annotation's classification. There are four classes of annotations. Any annotation generated by the modification of a variable name or value is classified as a Data Alteration. Any annotation generated by a merge operation is classified as Merge. Any annotation generated in relation to one of E-DataAid's import functions is assigned the Import class. All annotations not included in one of the other three classes are classified in the Comment class.



Class
Data Alteration
Merge
Comment

Source

The Source column displays a value of “USER” if the annotation was created by a user, and displays the value “SYSTEM” otherwise (i.e., for all automatically generated annotations).

Source
System
System
User

Location

The Location column displays where the annotation is stored. Annotations that are stored globally with the data file display the value “GLOBAL” in this column. Annotations saved with a particular session of data will indicate the subject number, session number, and experiment name for that session.

Location
10, 1, Tutorial
Global
Global

Reference

The Reference column references the value within a session to which this annotation pertains (e.g., if the annotation refers to the session’s experiment name, the reference is EXPNAME). If the annotation is stored globally, the reference is simply “Not applicable”.

Reference
SUBJECT
Not applicable
Not applicable

Text

The Text column displays the text included in the annotation.

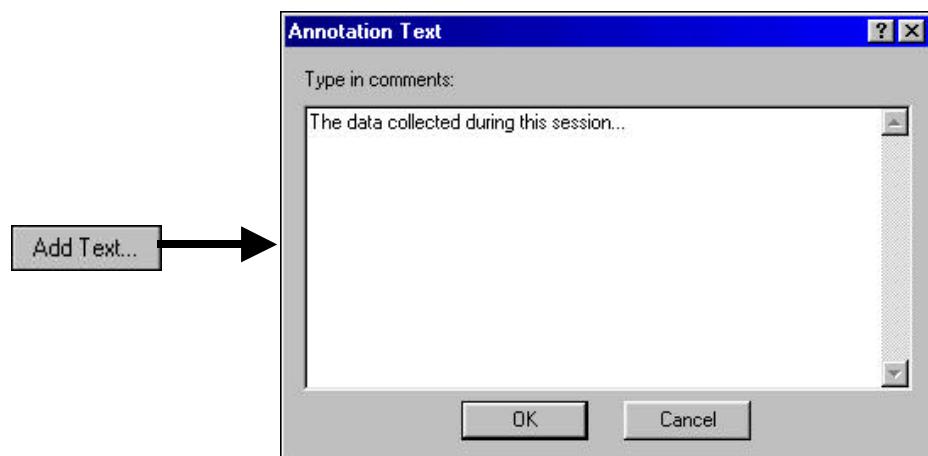
5.3.2.2 Adding Text to Annotations

Text may be added to an existing annotation using the Display Annotations command under the View menu, or the Display Annotations tool button. The Annotations dialog will be displayed, showing a list of all annotations included in the data file.



Time	Class	Source	Location	Reference	Text
11/5/98 10:49:06 AM	Merge	System	Global	Not applicable	Sessions merged from file C:\Program Files\Psychology Sof
11/5/98 10:49:06 AM	Merge	System	Global	Not applicable	Sessions merged from file C:\Program Files\Psychology Sof
11/5/98 10:49:06 AM	Merge	System	Global	Not applicable	Sessions merged from file C:\Program Files\Psychology Sof
11/5/98 10:49:06 AM	Merge	System	Global	Not applicable	Sessions merged from file C:\Program Files\Psychology Sof
11/5/98 10:49:06 AM	Merge	System	Global	Not applicable	Sessions merged from file C:\Program Files\Psychology Sof

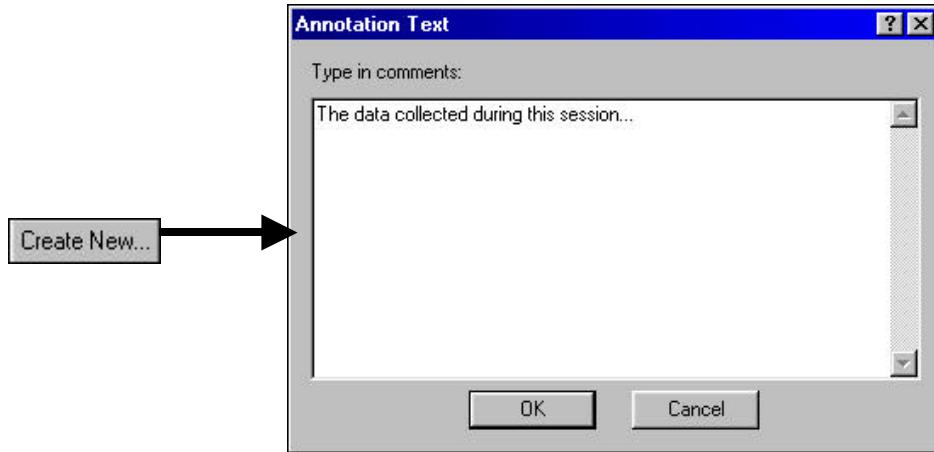
Text may be added to an existing annotation using the Add Text button at the bottom of the Annotations dialog. Select the annotation to which the text is to be added, and click the Add Text button. The Annotation Text dialog will be displayed allowing the user to enter text.



After the desired text has been added, click OK to update the annotation. Adding text to an annotation stores the text with the existing annotation, and updates the modified date for both the session and the data file.

5.3.2.3 Creating New Annotations

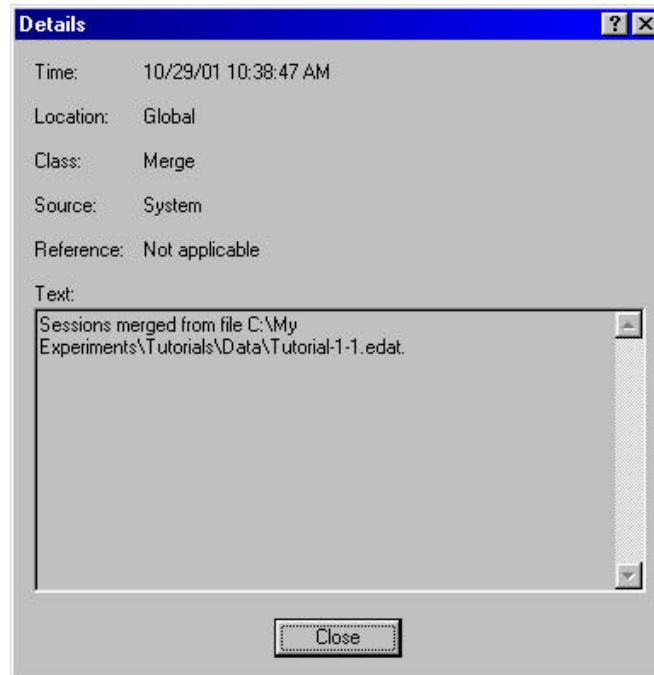
Any modifications made to a data file automatically generate an annotation describing the operation performed (i.e., renaming a variable, changing a value, etc.). Users may also create their own annotations (e.g., perhaps to make notes, or to mark a particular value). Annotations may be created using the Display Annotations command in the View menu or the Display Annotations tool button. The Annotations dialog will be displayed, showing a list of all annotations included in the data file. New annotations may be created using the Create New button at the bottom of the Annotations dialog. The Create New button displays the Annotation Text dialog that allows the user to enter text.



After entering the desired text, click OK to add the annotation to the data file. The creation of a new annotation stores the annotation globally, and updates the file's modified date.

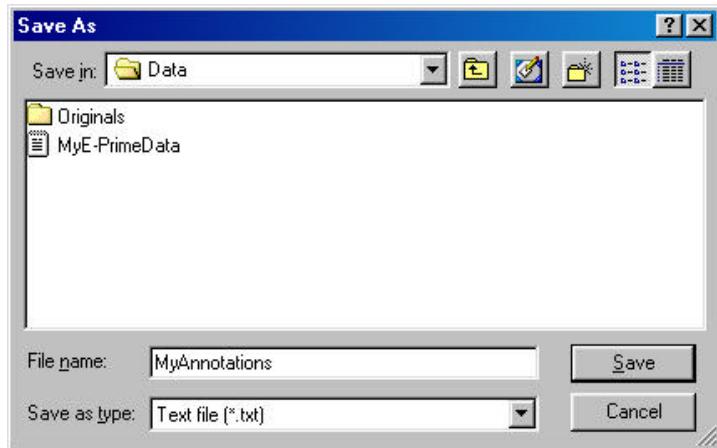
5.3.2.4 Viewing Details

The Details button in the Annotations dialog is used to display the entire text contained within an annotation. If an annotation's length exceeds the display area, select the annotation in the list, and click the Details button.



5.3.2.5 Exporting Annotations to Text

The Save As button on the Annotations dialog allows the user to save the annotations contained within the file to a text file.



The Save As command displays the Save As dialog. To save the annotations to a text file, enter the name of the file in which to save the annotations in the File name field and click Save.

5.3.3 Arranging the Spreadsheet

The spreadsheet displays the file's data in a grid format. When the application is first opened, no data is displayed until a specific data file is opened. When a data file is opened, the data is displayed in the spreadsheet with columns representing the variables of the experiment, and rows representing the lowest log level in the experiment (e.g., trials). The columns within the spreadsheet may be reorganized by moving, hiding, or unhiding columns.

	ExperimentName	Subject	Session	Date	Group	Time	Block	BlockList	BlockList.Cycle	BlockList.Sample	PracticeMode	Procedure[Block]
1	TutorialR9	1	1	10-26	1	09:23:	1	1	1	1	no	BlockProc
2	TutorialR9	1	1	10-26	1	09:23:	1	1	1	1	no	BlockProc
3	TutorialR9	1	1	10-26	1	09:23:	1	1	1	1	no	BlockProc
4	TutorialR9	1	1	10-26	1	09:23:	1	1	1	1	no	BlockProc
5	TutorialR9	1	1	10-26	1	09:23:	1	1	1	1	no	BlockProc
6	TutorialR9	1	1	10-26	1	09:23:	1	1	1	1	no	BlockProc
7	TutorialR9	1	1	10-26	1	09:23:	1	1	1	1	no	BlockProc
8	TutorialR9	1	1	10-26	1	09:23:	1	1	1	1	no	BlockProc
9	TutorialR9	2	1	10-26	1	09:24:	1	1	1	1	no	BlockProc
10	TutorialR9	2	1	10-26	1	09:24:	1	1	1	1	no	BlockProc
11	TutorialR9	2	1	10-26	1	09:24:	1	1	1	1	no	BlockProc
12	TutorialR9	2	1	10-26	1	09:24:	1	1	1	1	no	BlockProc
13	TutorialR9	2	1	10-26	1	09:24:	1	1	1	1	no	BlockProc
14	TutorialR9	2	1	10-26	1	09:24:	1	1	1	1	no	BlockProc
15	TutorialR9	2	1	10-26	1	09:24:	1	1	1	1	no	BlockProc
16	TutorialR9	2	1	10-26	1	09:24:	1	1	1	1	no	BlockProc
17	TutorialR9	3	1	10-26	1	09:24:	1	1	1	1	no	BlockProc
18	TutorialR9	3	1	10-26	1	09:24:	1	1	1	1	no	BlockProc
19	TutorialR9	3	1	10-26	1	09:24:	1	1	1	1	no	BlockProc
20	TutorialR9	3	1	10-26	1	09:24:	1	1	1	1	no	BlockProc
21	TutorialR9	3	1	10-26	1	09:24:	1	1	1	1	no	BlockProc
22	TutorialR9	3	1	10-26	1	09:24:	1	1	1	1	no	BlockProc
23	TutorialR9	3	1	10-26	1	09:24:	1	1	1	1	no	BlockProc
24	TutorialR9	3	1	10-26	1	09:24:	1	1	1	1	no	BlockProc
25												

5.3.3.1 Spreadsheet Organization

By default, ExperimentName, Subject, and Session appear as the first three variable columns. The remaining variables are organized alphabetically within hierarchical log levels. That is, session level variables occur in alphabetical order after the first three columns, then the variables for the next log level (e.g., Block) are organized in alphabetical order, then the variables for the



next log level (e.g., Trial) are organized in alphabetical order, etc. The default column order is used every time a file is opened (i.e., any reordering of columns is not saved when a file is closed), or when the Restore Column Order or Restore Spreadsheet commands are used.

5.3.3.2 Selecting Columns

Using a Mouse

A column may be selected by clicking on the column heading or by selecting a cell within the desired column and using the Select Column command in the Edit menu.

Using the Keyboard

A column may be selected using the keyboard by using the arrow keys to navigate to a cell within the desired column, and simultaneously pressing the **Ctrl** and **L** keys.

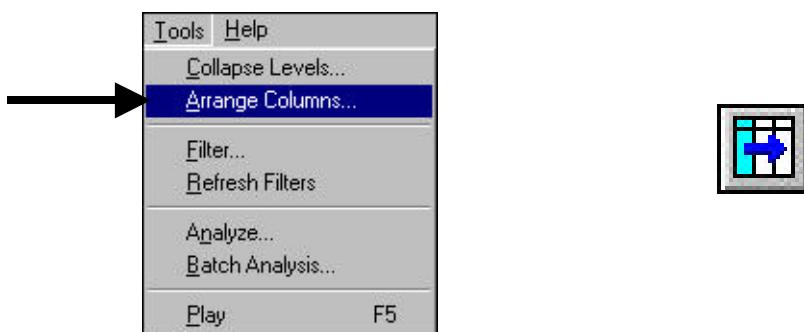
5.3.3.3 Resizing Columns

When the E-DataAid application is opened, the columns are sized to display the entire column header. To resize the width of a column, drag the boundary on the right side of the column header to increase or decrease the width of the column to the desired width. Double clicking the boundary on the right side of the column header will expand the column to display the larger of either the column header or the data contained within the column's cells. Multiple columns may be simultaneously sized to a standard width by selecting a range of columns, and dragging the boundary for one of the selected columns to the desired width.

5.3.3.4 Arranging Columns

By default, the columns in a data file are displayed in alphabetical order within hierarchical levels (e.g., Session, Block, Trial, etc.). Columns may be rearranged by selecting the column, clicking and holding the left mouse button to change the cursor to an arrow on a small box, dragging the column header to a new location, and releasing the left mouse button. A dotted red line provides feedback concerning where the column will be located after the left mouse button is released.

Alternatively, the Arrange Columns dialog may be displayed using the Arrange Columns tool button on the toolbar, or by using the Arrange Columns command in the Tools menu.

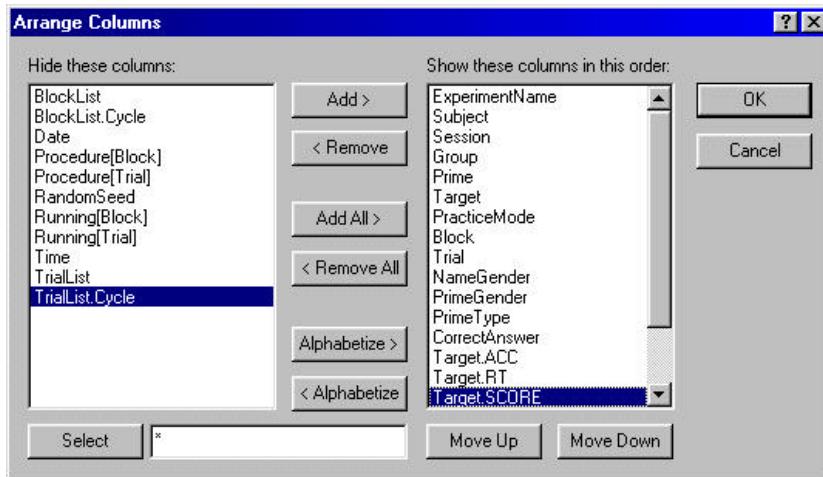


The Arrange Columns dialog box allows arranging columns within the list of currently displayed columns by selecting the column name and using the Move Up and Move Down buttons to move the column to a new position.



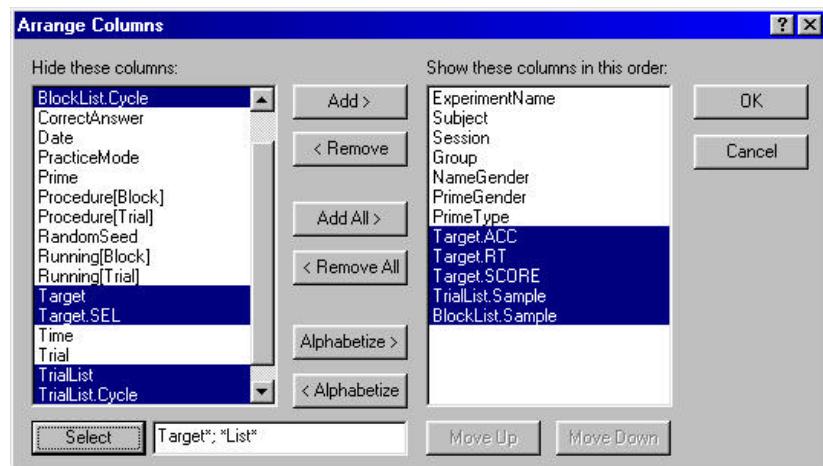
E-Prime Reference Guide

Chapter 5: E-DataAid



The Arrange Columns dialog also allows the reorganization of the spreadsheet through the designation of which columns to hide or show. To hide a column so that it does not appear in the spreadsheet, select the column in the Show list and click the Remove button. The column will then be placed in the Hide column. Likewise, to display any hidden columns in the spreadsheet, select the column in the Hide list and click the Add button to add the column to the Show list. When a hidden column is added to the Show list, the column is added to the end of the Show list, and is displayed as the last column in the spreadsheet when the dialog is dismissed. Use the Move Up button to reposition the variable if desired. The Add All button moves all hidden columns (selected or not) to the Show list. The Remove All button moves all unhidden columns (selected or not) to the Hide list. The Alphabetize buttons allow the items in either list to be ordered alphabetically.

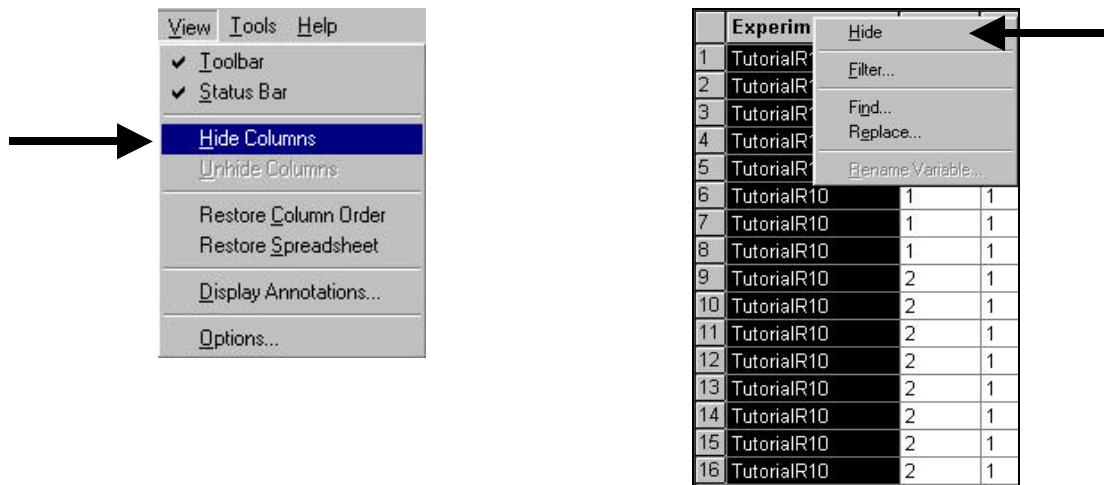
The Select field may be used to highlight columns in both lists matching a specific filter. In the Select field below the hidden columns list, type the desired filter into the edit box and click the Select button. Any column in either list matching the filter will be highlighted. The highlighted column then may be added or removed from the respective list. The selection is not case sensitive, and multiple filters may be entered at one time. Multiple filters are separated using semi-colons. Filters may also include the wildcard (*) character.



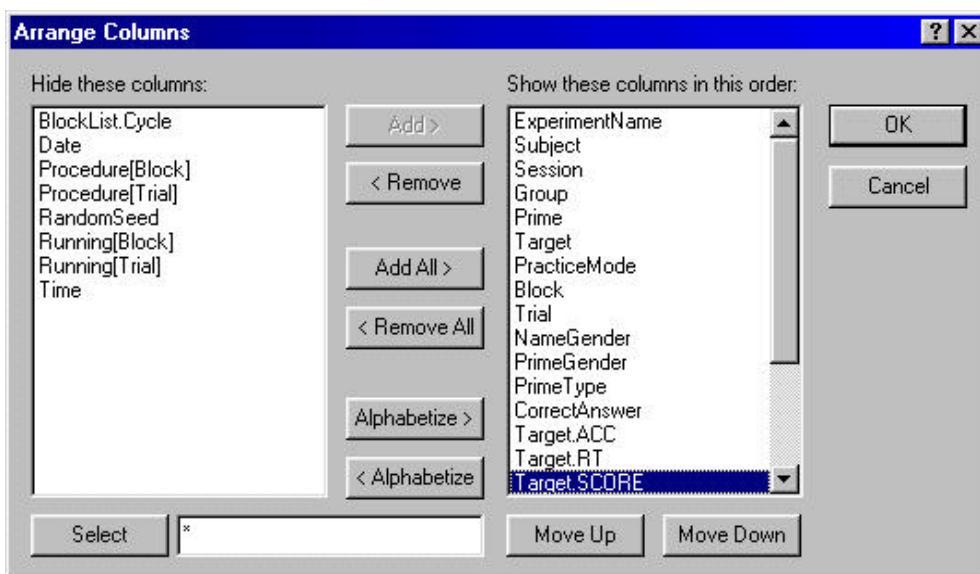


5.3.3.5 Hiding/Unhiding Columns

Columns may be hidden or unhidden using a variety of methods. First, columns may be hidden by selecting the column(s) to hide and using the Hide Columns command in the View menu. Or, individual columns may be hidden by right-clicking the column header and choosing the Hide command from the pop-up context menu.



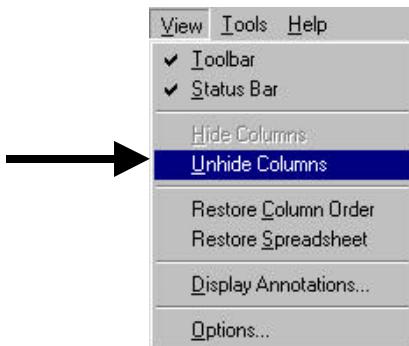
Columns may also be hidden using the Arrange Columns command in the Tools menu. Using this method, the Arrange Columns dialog box will be opened to display all of the columns currently displayed or hidden, and the order of the displayed variables. Columns may be hidden by clicking the column name(s) and using the Remove button to move them from the list of displayed columns to the "Hide these columns list."



Finally, columns may be hidden by dragging the boundary on the right side of the column header to meet the boundary on the left side of the header.



There are also several ways to unhide columns. To unhide one or more columns, use the Arrange Columns command in the Tools menu. The Arrange Columns dialog box will be opened to display all of the columns currently displayed or hidden. Unhide a column by selecting the column name and using the Add button to move the column from the hidden columns list to the Show list. The unhidden column may be positioned in the list of displayed columns using the Move Up or Move Down buttons. To unhide all hidden columns, use the Add All button in the Arrange Columns dialog, or the Unhide Columns command in the View menu.



When using the Add All button in the Arrange Columns dialog to unhide all hidden columns, the hidden columns are placed at the end of the Show list, and are placed as the last columns in the spreadsheet. Use the Move Up or Move Down buttons in the Arrange Columns dialog to reposition the columns if desired. When unhidng columns with the Unhide Columns command, the columns are shown in their original placement in the spreadsheet.

5.3.3.6 Restoring Column Order

The Restore Column Order command in the View menu restores the columns to the default order. The spreadsheet's default order is as follows:

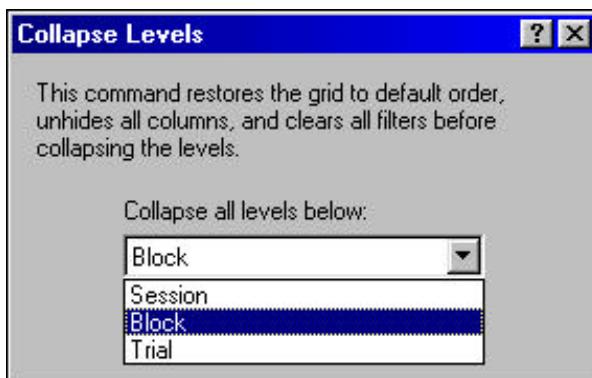
- ExperimentName
 - Subject
 - Session
 - Session level variables in alphabetical order
 - Name of the second level (e.g., Block)
 - Second level variables (e.g., block level variables) in alphabetical order
 - Name of third level (e.g., Trial)
 - Third level variables (e.g., trial level variables) in alphabetical order
- ...etc.

5.3.3.7 Collapsing Levels

The Collapse Levels command allows the user to collapse all levels below a chosen level. Collapsing all levels below a designated level automatically restores the columns to the default order, redisplays all hidden columns, clears all filters, filters all levels below the designated level on instance 1 or NULL, and hides all columns below the designated level. The Collapse Levels command may be invoked using the Tools menu, or by clicking the Collapse Levels tool button.



The Collapse Levels command displays the Collapse Levels dialog. By default, the spreadsheet is set to collapse levels below the lowest level of data (e.g., commonly the lowest level would be the Trial level). Use the drop-down box to select a level, and click OK. The spreadsheet will be refreshed to display all data at or above the selected level.



5.3.3.8 Restoring Spreadsheet

The Restore Spreadsheet command, in the View menu, clears all active filters, redisplays all hidden columns, and restores the columns to the default order. The spreadsheet's default order is as follows:

- ExperimentName
 - Subject
 - Session
 - Session level variables in alphabetical order
 - Name of the second level (e.g., Block)
 - Second level variables (e.g., block level variables) in alphabetical order
 - Name of third level (e.g., Trial)
 - Third level variables (e.g., trial level variables) in alphabetical order
- ...etc.

5.3.4 Closing a Data File

A data file may be closed without completely exiting the E-DataAid application. To close a data file, use the Close command in the File menu. If the open file has been modified, the application will prompt the user to save before closing the file.



5.3.5 Editing Data

E-DataAid supports editing of the values within a data file, but does not function exactly like other spreadsheet applications. Specifically, due to the hierarchical nature of experimental data, E-DataAid structures the spreadsheet to maintain the hierarchical relationship between data collected at higher and lower levels (e.g., block level, trial level, etc.).

Data from one level in an experiment is hierarchically related to the data in other levels. Higher level data is shared by many trials (e.g., the same subject number applies to all trials run by that subject). However, spreadsheets require a flat grid format. Thus, because the rows in E-DataAid's spreadsheet represent the lowest level of data in the data file, E-DataAid must repeat the data values collected for higher levels down many rows of the spreadsheet.

In order to maintain the hierarchical nature of the data while displaying in flat grid format, E-DataAid creates pointers from the individual cells to the piece of data they display. Cells containing lower level data (e.g., trial level data) may share upper level data (e.g., block or session level data), such that many cells are pointing to the same piece of data. For this reason, when an individual cell is edited, all other cells pointing to the piece of edited data will automatically be modified when the new value is accepted in the edited cell.

In addition, to help preserve the integrity of the data, E-DataAid keeps an audit trail of any modifications made to a file. For each modification to a file, a comment describing the modification is written as an annotation to the file. When a value is edited, the application stores the annotation with the session containing the modified value, updates the session's modified date, and updates the file's modified date.

E-DataAid allows modification of the experiment name, subject or session number, individual cell values, variable names, or level names. In addition, variables may be added, annotations may be created, and text may be added to existing annotations. If a cell cannot be edited, it will appear in gray, and the cursor cannot be placed in the cell. Any cell in a column displaying level numbers for any level below the session level will be read-only (i.e., block numbers, trial numbers, etc.). Because of the hierarchical nature of experiment data, the application does not allow these cells to be edited. Any cell in a column on which an Administrator places a security restriction will also be read-only.

Each modification writes an annotation describing the modification to the data file. The image below illustrates an Annotations record for a file containing a sequence of file merges, as well as a data alteration (i.e., changing the subject number).

The screenshot shows a Windows-style dialog box titled "Annotations". Inside, a message says: "To view truncated text, select the annotation in the list and click the Details button." Below is a table with columns: Time, Class, Source, Location, Reference, and Text. The table lists six entries:

Time	Class	Source	Location	Reference	Text
1/31/00 11:36:05 AM	Data Alteration	System	11, 1, Tutorial	SUBJECT	Changed subject number - Old: 1 New: 11.
12/21/99 10:34:36 AM	Merge	System	Global	Not applicable	Sessions merged from file C:\My Experiments\Tuto
12/21/99 10:34:36 AM	Merge	System	Global	Not applicable	Sessions merged from file C:\My Experiments\Tuto
12/21/99 10:34:36 AM	Merge	System	Global	Not applicable	Sessions merged from file C:\My Experiments\Tuto
12/21/99 10:34:36 AM	Merge	System	Global	Not applicable	Sessions merged from file C:\My Experiments\Tuto
12/21/99 10:34:36 AM	Merge	System	Global	Not applicable	Sessions merged from file C:\My Experiments\Tuto

At the bottom are buttons: Show: All, Details..., Add Text..., Create New..., Save As..., and Close.



For each value modified, the application stores an annotation with the session that contains the modified value, and updates the session's modified date. All other annotations are stored globally in the data file. All modifications result in the updating of the file's modified date.

In the spreadsheet, all data alterations, including edits to a cell value or a variable name, are displayed in color. By default, data alterations are displayed in red.

Subject
12
1
1
1
1
1
1
1
1

The display color for data alterations may be changed to blue or green by using the Options command in the View menu. Data alterations result in the marking of the file as containing data alterations.

5.3.5.1 Editing Experiment Name

The experiment name may be edited with the restriction that every session must have an experiment name, the experiment name cannot be white space, and it cannot exceed 80 characters in length. To edit a session's experiment name, select a cell in the ExperimentName column for any row containing data for that session. The application will acknowledge the edit when the Enter key is pressed, the cell is exited, or the file is saved.

Each data file may contain sessions from different experiments, and each session stores the name of the experiment to which it is related. Therefore, when an experiment name is changed for one session of data, only the cells related to that session are modified. To change the experiment name for more than one session at a time, use the Replace command in the Edit menu.

A file can contain data sessions from more than one experiment. Therefore, when a session's experiment name is modified, the application checks for values which will result in a duplicate subject number, session number, and experiment name combination. If the combination of subject number, session number and experiment name is duplicated in the data file, the application will display a warning, and the user will be given the opportunity to cancel the operation.

Editing a session's experiment name will mark the file as containing altered data, will store an annotation with the session, and will update the modified dates for both the session and the data file containing that session. Modified dates may be viewed in the file's properties.

5.3.5.2 Edit Subject and Session Number

Subject and Session numbers may be modified with the restrictions that every session of data must have a subject and session number, and that these numbers must be positive integers (up to 32,767). To edit a subject or session number, type the new number in one of the cells in the appropriate column (Subject or Session column) containing the value for the change. The value



will not be accepted until the {Enter} key is pressed, until the cursor is placed in another cell, or until the file is saved. When the new value is accepted, all of the cells to which the modification applies will be changed to display the new value, and the edited data will be displayed in red.

A file can contain more than one experiment. Therefore, when subject or session number is modified, the application checks for values which will result in a duplicate subject number, session number, and experiment name combination. If the combination of subject number, session number and experiment name is duplicated in the data file, the application will display a warning, and the user will be given the opportunity to cancel the operation.

Editing subject and/or session number will mark the file as containing altered data, will store an annotation with the session, and will update the modified dates for both the session and the data file containing that session. Modified dates may be viewed in the file's properties.

5.3.5.3 Editing Cell Values

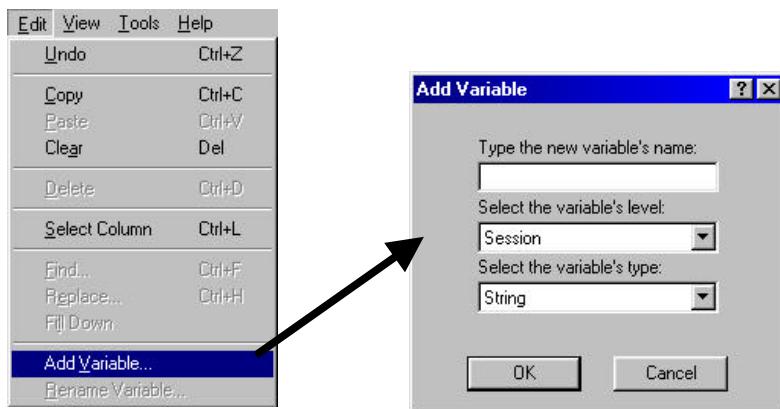
Individual cell values may be modified with certain restrictions. Integer and float variables may only be modified with number input, the application truncates any decimal point and places for integer types, and the application does not check for data values that are too small or too large for the defined type. To edit the value in a cell, type the new value directly in the cell. The value will not be accepted until the Enter key is pressed, until the cursor is placed in another cell, or until the file is saved.

To "delete" the value in a cell, use the Delete command in the Edit menu, or simultaneously press the Ctrl and D keys when the appropriate cell is selected. The application requires this method because data may not actually be deleted. The Delete command sets an internal flag for the selected piece of data to missing.

Editing a cell value will mark the file as containing altered data, will store an annotation with the session, and will update the modified dates for both the session and the data file containing that session. Modified dates may be viewed in the file's properties.

5.3.5.4 Adding Variables

Variables may be added using the Add Variable command in the Edit menu. The Add Variable command displays a dialog allowing the user to define a new variable's name, its level (session, block, trial), and its data type (string, integer, float). The new variable will be added as a new column at the end of the spreadsheet.



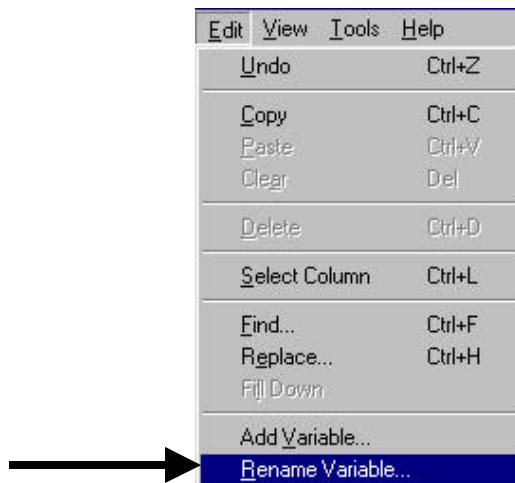


Adding a variable to a data file marks the file as containing altered data, displays the new variable in color, stores a global annotation describing the addition of the new variable, and updates the modified date for the file. Variables may be added to a data file with the restrictions that the name of the variable must be fewer than 80 characters and must begin with a letter, the name may only include valid characters (letter, numbers, periods and underscores), and the name cannot be "Subject", "Session", "ExperimentName", or the name of any other existing variable or level.

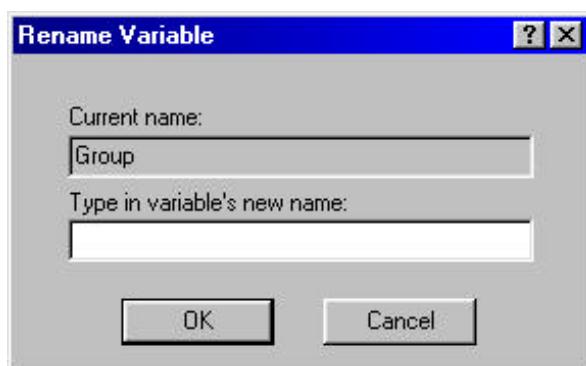
5.3.5.5 Renaming Variables

Variables may be renamed with the restrictions that the name of the variable must be fewer than 80 characters and must begin with a letter, the name may only include valid characters (letter, numbers, periods and underscores), and cannot be "Subject", "Session", "ExperimentName", or the name of any other existing variable or level.

Variables and level names may be edited using the Rename Variable command in the Edit menu.



The Rename Variable command displays a dialog allowing the user to enter a new name for the variable or level. When a new name is entered, the application will search the data file, and will display an error if the new name is already used by another variable or level in the file. The search is not case sensitive.





Editing a variable or level name marks the data file as containing altered data, displays the edited variable or level name in color, stores a global annotation describing the edit, and updates the modified date for the file.

5.3.5.6 Copying Data

The Copy command in the Edit menu copies the contents of the currently selected cell, cells, rows, or columns to the clipboard. This information may then be pasted (using the Paste command from the Edit menu) into a selected area within the spreadsheet, or into another application.

5.3.5.7 Pasting Data

The Paste command in the Edit menu pastes the contents of the clipboard into the selected cell, cells, or column. Only three types of paste functions are supported.

1. One piece of data from the clipboard to one cell.
2. One piece of data from the clipboard to a range of cells one column wide.
3. One column of data from the clipboard to a range of cells one column wide.

With the exception of number two, the copy area and the paste area must be the same size and shape.

5.3.5.8 Fill Down

The Fill Down command in the Edit menu fills all cells in the selected range of cells within a single column with the contents of the first cell in the range. This command is only available for editable cells. Select a range of cells by clicking in the first cell and dragging down the column to highlight the desired area. With the desired area highlighted (Figure 1), select the Fill Down command from the Edit menu (Figure 2). The value from the first selected cell will be filled into the other cells in the selected area (Figure 3).

Color
blue

Figure 1: Area highlighted for Fill Down

Edit	View	Tools	Help
Undo		Ctrl+Z	
Copy		Ctrl+C	
Paste		Ctrl+V	
Clear		Del	
Delete		Ctrl+D	
Select Column		Ctrl+L	
Find...		Ctrl+F	
Replace...		Ctrl+H	
Fill Down			
Add Variable...			
Rename Variable...			



Figure 2: Fill Down command in Edit menu

Color
blue

Figure 3: Fill Down complete

5.3.5.9 Deleting Data

The Delete command in the Edit menu designates the selected cells as having missing data (NULL). The Delete command is only available for cells containing data for variables that may be edited. To designate the value of a cell as missing data, use the Delete command. The Delete command sets the underlying data to NULL. Entering an empty string in a cell or actually typing

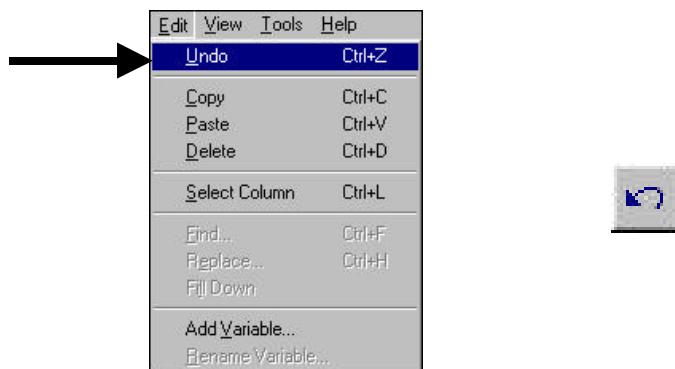


the word “NULL” into the cell sets the underlying data to the string value “” or “NULL”, respectively.

It is important to distinguish between deleted values and values set to the string “NULL”. Typing the word “NULL” sets the underlying data to the string value “NULL”. Similarly, copying a cell with no data to another cell does not delete the data in the other cell, but rather sets its value to the string value “NULL”. In order to determine whether a value has actually been deleted or is set to the string value “NULL”, view the file’s annotations. If the value was deleted, NULL will appear in the annotation without quotes. If the value was set to the string value “NULL” in quotes.

5.3.5.10 Undoing Edits

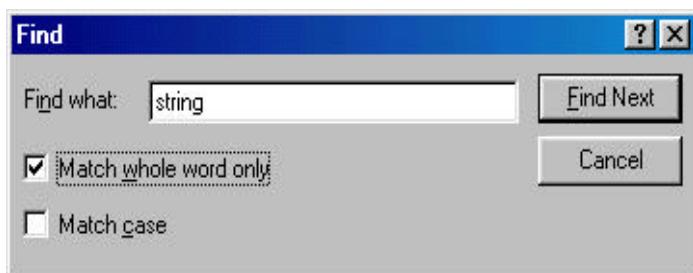
E-DataAid supports undoing all modifications made during the current E-DataAid session. The most current modification may be undone by using the Undo command in the Edit menu, or the Undo tool button on the toolbar.



To undo previous modifications, continue to use the Undo command until the desired operation is reversed. If no modifications have been made to the file (i.e., there is nothing to undo), the Undo button and command are disabled.

5.3.5.11 Finding Data

The Find command in the Edit menu allows the user to search a range of cells, columns, or rows for specified text. Select an area to be searched by clicking and dragging the mouse across the desired area to highlight it. With the desired area highlighted, select the Find command from the Edit menu. The Find dialog is displayed.



The text to be located is entered into the “Find what” field. The Find Next button locates the next occurrence of the string in the selected area. The Match whole word only box is checked by default, and searches for the exact word in the Find what field. The Match case box (unchecked



by default) requires the search to match the case of the characters within the string as well as the actual text.

5.3.5.12 Replacing Data

The Replace command in the Edit menu displays the Replace dialog, which allows the user to search for specific text within a range of cells, columns or rows, and replace the text with a specified string. To replace one string with another, select the desired range of cells, columns or rows by clicking and dragging across the desired area. With the area highlighted, select the Replace command from the Edit menu, or right click on the highlighted area to display the pop-up context menu, and select Replace.

The image shows two windows side-by-side. On the left is a screenshot of the 'Edit' menu. An arrow points from the 'Edit' menu towards a context menu on the right. The context menu is displayed over a table row. The 'Replace...' option in both the 'Edit' menu and the context menu is highlighted with a blue selection bar.

Edit View Tools Help

Undo Ctrl+Z

Copy Ctrl+C

Paste Ctrl+V

Clear Del

Delete Ctrl+D

Select Column Ctrl+L

Find... Ctrl+F

Replace... Ctrl+H

Fill Down

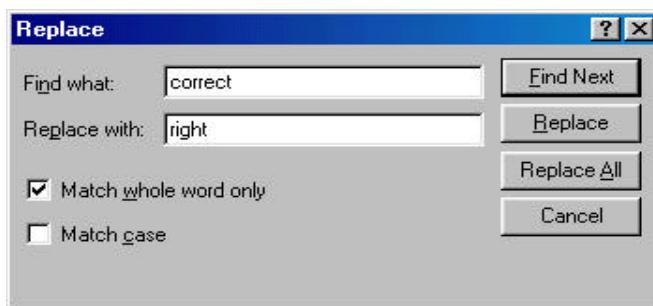
Add Variable...

Rename Variable...

Target SCORE | **Target SEL**

corre	Hide
wron	Filter...
corre	Find...
corre	Replace...
corre	Rename Variable...
correct	2
correct	1
correct	2
wrong	2
correct	1
correct	1
correct	2
correct	2
correct	1
wrong	1
correct	1

The Replace command displays the Replace dialog. In the "Find what" field in the Replace dialog, the user specifies the string for which to search, and the string to which to change any instance of that value is specified in the "Replace with" field.

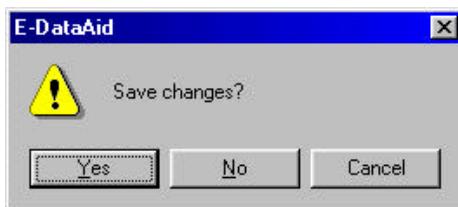


The Find Next, Replace, and Replace All buttons allow the user to search for and replace each instance of a specific string individually or for every instance within the specified area. To search for a string, or the next instance of a string, specify the string in the "Find what" field and click the Find Next button. When the string is located, the search will pause for further direction. To replace the string with another string, enter a value in the "Replace with" field and click the Replace button. To replace all instances of a string within a specified area, use the Replace All button. The Match whole word only box is checked by default, and searches for the exact word in the Find what field. The Match Case box (unchecked by default) requires the search to match the case of the characters within the string as well as the actual text.



5.3.6 Exiting E-DataAid

The E-DataAid application may be exited using the Exit command in the File menu. If the opened file is untitled or has been modified, the application will prompt the user to save before exiting.



5.3.7 Exporting

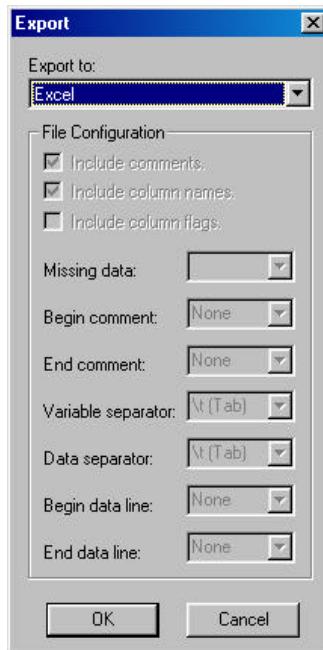
E-Prime data files may be exported to a tab-delimited text file for import into an external package, or for re-import into E-DataAid via the Export command in the File menu or the Export tool button.



The Export command displays the Export dialog box, which allows the export of all **unhidden columns and rows in the spreadsheet** to a text file. The dialog offers options to export to a format compatible with Excel, StatView, SPSS, or E-Prime text. The dialog also offers an "Other" option, which allows some flexibility in formatting the text file so that it can later be imported into other packages with minimal modifications.

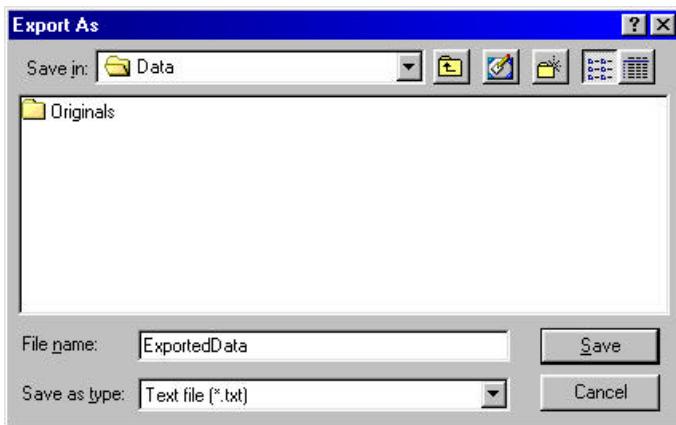
5.3.7.1 Export Data to Excel

To export to Excel, choose the Excel option in the "Export to" field. When exporting to Excel, the application automatically chooses the file configuration to export to a tab-delimited text file.



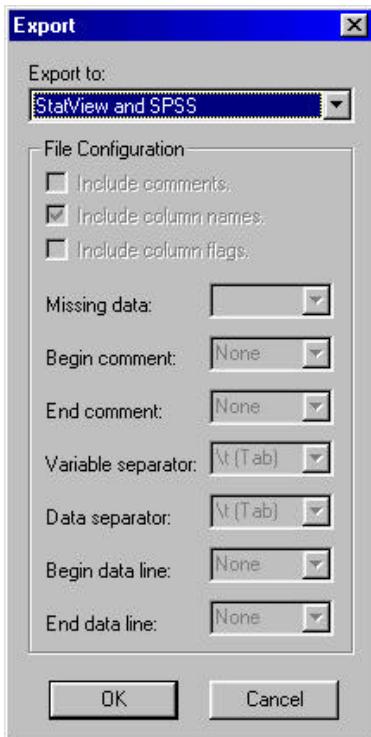


After the format for export is specified, the Export As dialog is displayed. In the Export As dialog, navigate to the path where the file is to be saved, and supply a name for the file in the “File Name” field. Once exported, the resulting text file can be imported into Excel using the Open command in the Excel File menu.



5.3.7.2 Export Data to StatView

To export to StatView or SPSS, choose the StatView and SPSS option in the “Export to” field. When exporting to StatView, the application automatically chooses the file configuration to export to a tab-delimited text file.



After the format for export is specified (click OK in the Export dialog), the Export As dialog is displayed. In the Export As dialog, navigate to the path where the file is to be saved, and supply

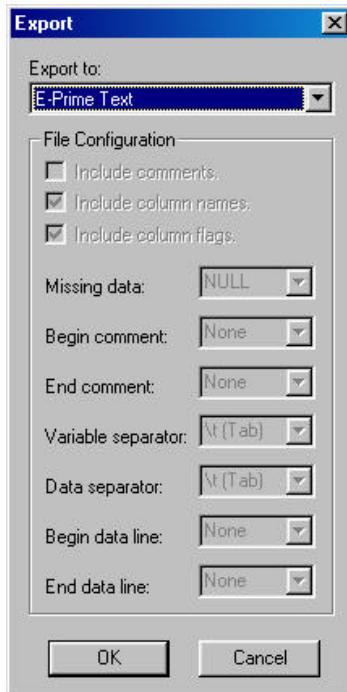


a name for the file in the "File name" field. Once exported, the resulting text file can be imported into Statview using the Open command in the StatView File menu.

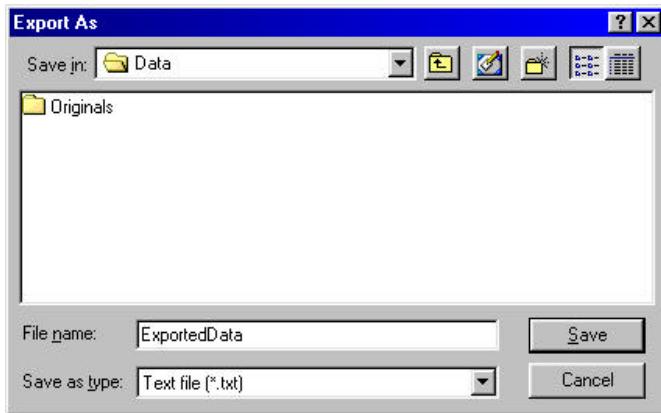


5.3.7.3 Export Data to E-Prime Text

To export to a text file that may be re-imported into E-DataAid, choose the E-Prime Text option in the "Export to" field. When exporting to E-Prime text, the application automatically chooses the file configuration to export to a tab-delimited text file.



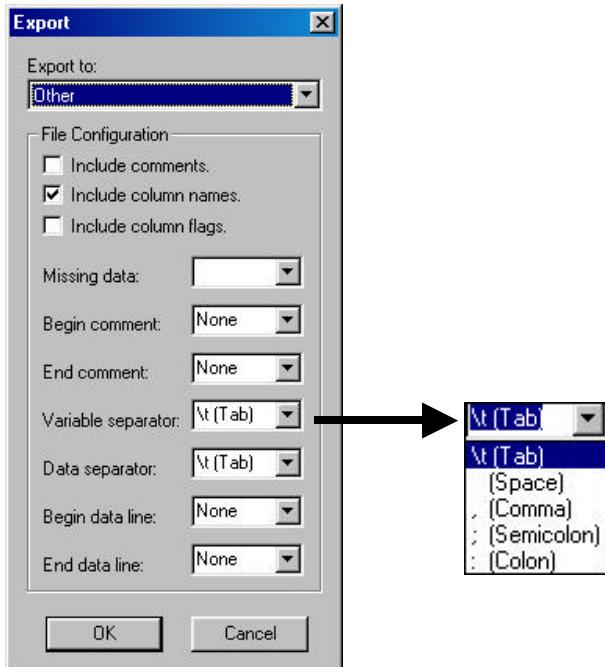
After the format for export is specified (click OK in the Export dialog), the Export As dialog is displayed. In the Export As dialog, navigate to the path where the file is to be saved, and supply a name for the file in the File Name field.



Once exported, the resulting text file can be re-imported into E-DataAid using the Import command in the E-DataAid File menu. Note: only data visible in the spreadsheet will be exported. In order to export the raw data file as E-Prime text, redisplay all hidden rows and columns and remove active filters, or use the Restore Spreadsheet command. The Restore Spreadsheet command in the View menu clears all active filters, redisperses all hidden columns, and restores the columns to the default order.

5.3.7.4 Export Data to Other Format

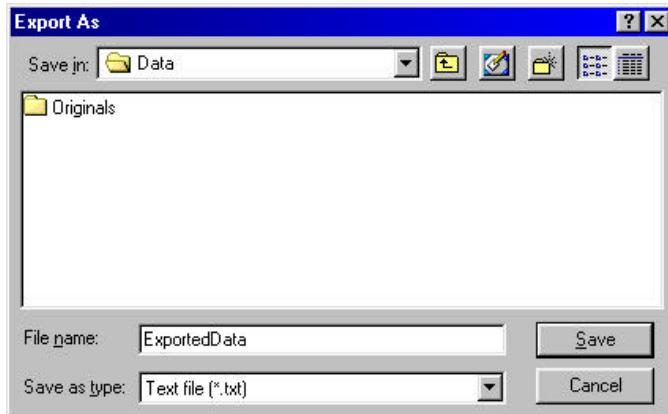
To export to a text file which may be imported into an external package other than StatView or Excel, choose the "Other" option in the "Export to" field. When exporting using the "Other" option, the application allows the user to designate the file configuration for the export.



The user must designate how to export missing data, comments, and column headers. In addition, the configuration must indicate how the separators between variables and data are specified, as well as the information indicating the beginning and ends of individual lines of data.



Common characters appropriate for each field are included in drop-down boxes for each field. The user may select from one of the options in the drop-down box, or may type another alternative directly in the field. After the format for export is specified (click OK in the Export dialog), the Export As dialog is displayed. In the Export As dialog, navigate to the path where the file is to be saved, and supply a name for the file in the File Name field. Once exported, the resulting text file can be imported into the intended external package.

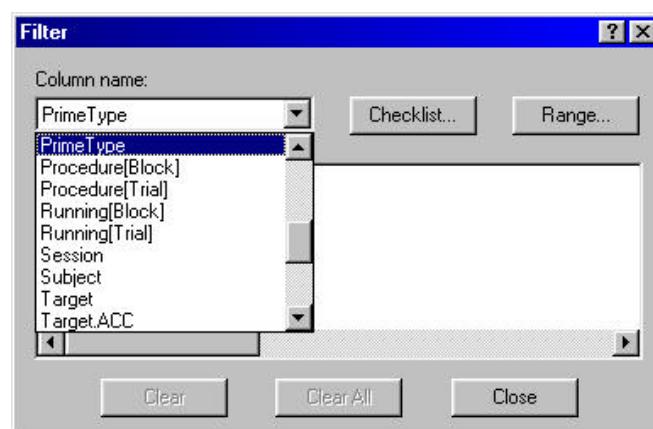


5.3.8 Filtering

E-DataAid allows the application of filters to one or more columns in the spreadsheet. A Checklist filter and a Range filter may be applied to each column. The Filter command displays the Filter dialog box, which allows the user to apply filters to the spreadsheet or clear active filters. The Filter command is invoked through the Tools menu, by right clicking a column header and choosing the Filter option, or by double clicking a column header.

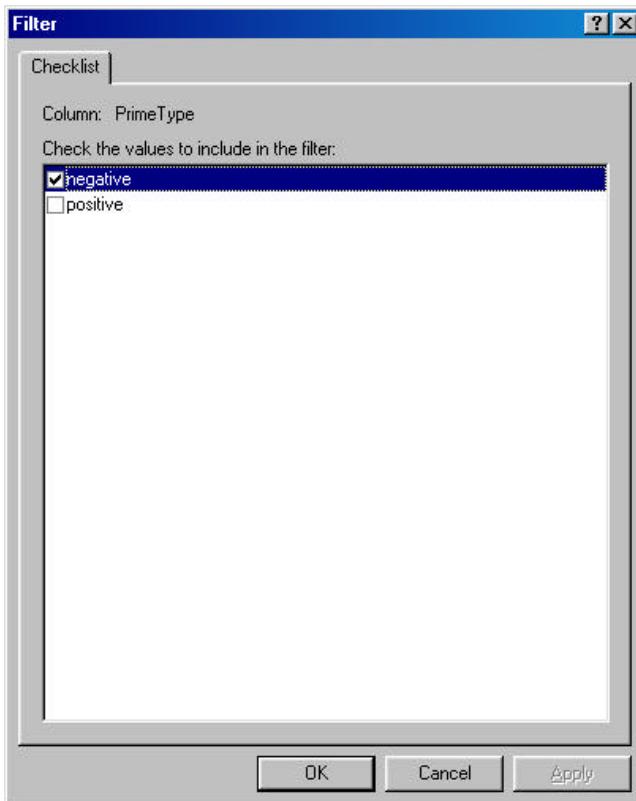
5.3.8.1 Checklist Filter

Checklist filters may be used to restrict the data set up to 32,767 values. The Checklist tab on the Filter dialog displays a list of unique values contained within the data set. To apply a checklist filter to a column, click the Filter button on the Analyze dialog. The Filter dialog will be displayed, allowing the selection of the column to be filtered, as well as the type of filter to be applied. In the "Column name" field on the Filter dialog, use the combo box to select the name of the column to be filtered, and click the Checklist button.





Checklist filters are inclusive such that only those values checked in the list are included in the data set for analysis. To include a value, check the box next to the value.



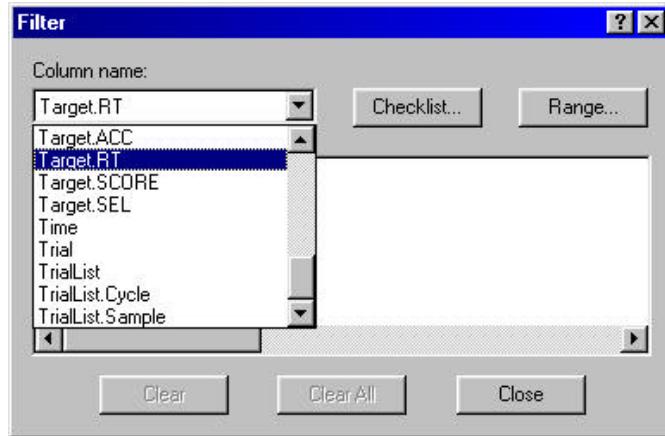
Checklist filters are immediately applied to the spreadsheet when the OK button is clicked. When a checklist filter is applied, only those values for which boxes were checked will be displayed. If no boxes are checked, the application assumes no filter is applied, and displays all values.

Multiple values may be selected by simply checking all boxes next to values to be included in the filter. To select a consecutive group of values, click the first value in the group with the mouse, hold down the Shift key, and click the last value in the group. All values between the first and the last will be highlighted. To highlight a non-consecutive group of values, hold the Ctrl key down while clicking all desired values with the mouse. Checking or unchecking any value in a selected group will check or uncheck all of the values in the group while the group is highlighted.

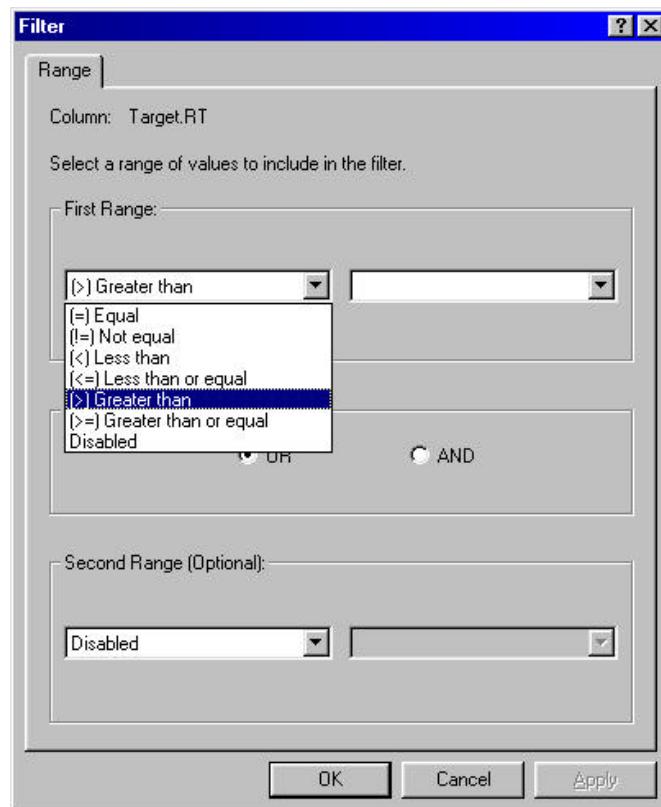
5.3.8.2 Range Filter

Range filters may be used to restrict the data set to a specific range of values. The Range tab on the Filter dialog allows the setting of up to two filters to the selected column. When more than one range filter is applied to a column, an operator (either AND or OR) is required.

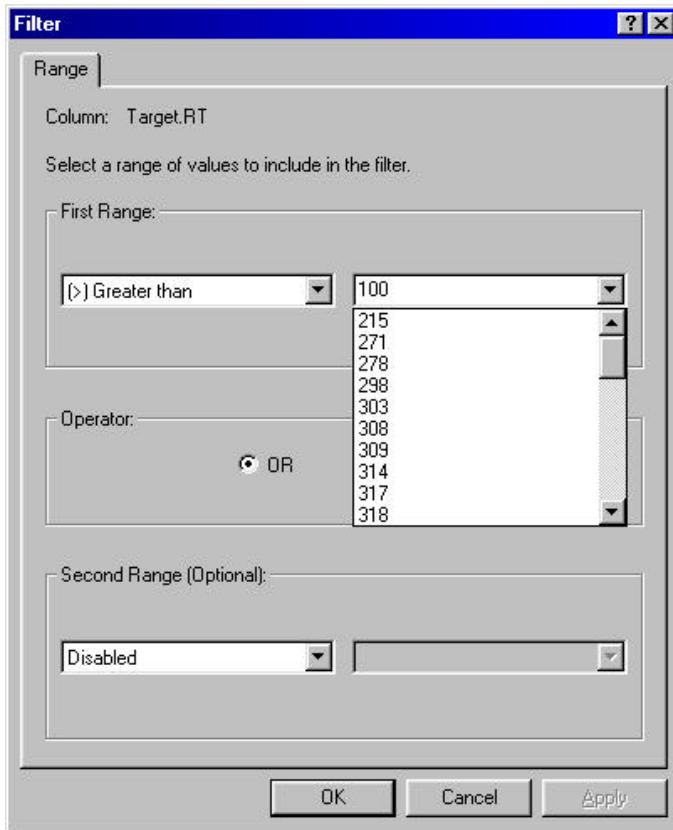
To apply a range filter to a column, click the Filter button on the Analyze dialog. The Filter dialog will be displayed allowing the selection of the column to be filtered, as well as the type of filter to be applied. In the "Column name" field on the Filter dialog, use the combo box to select the name of the column to be filtered, and click the Range button.



Range filters may be applied using mathematical operations available from the combo box on the left side of the First Range or Second Range fields in the Filter dialog. Operations include Equal (=), Not equal (!=), Less than (<), Less than or equal (<=), Greater than (>), Greater than or equal (>=) and Disabled (no filter).



The mathematical operations may be applied to the range of values contained within the selected variable column. The range of unique values (up to 32,767) within the column may be viewed using the drop-down list associated with the edit box on the right side of the First Range or Second Range fields. If no drop-down box is available, modify the Range Filter option via the Options command on the View menu.



In the First Range field on the Range page, specify the range to be applied in the first filter. A second Range filter may also be applied to the same column if desired (using the AND or OR operators).

Range filters are inclusive such that only those values within the specified range(s) will be included in the data set for analysis. Range filters are immediately applied to the spreadsheet when the OK button is clicked. When a range filter is applied, only those values included within the specified range(s) will be displayed in the spreadsheet. For example, to include only reaction times greater than 100 msec in the analysis, a range filter may be applied using the “(>) Greater than” operation. To filter for reaction times between 100 and 1000 msec, the AND operator would be used to set “(>) Greater than 100” in the First Range field and “(<) Less than 1000” in the Second Range field.

5.3.8.3 Display of Active Filters

Active filters are displayed in the Filters bar at the bottom of the application display. If there are no filters, the Filters bar's display area remains blank. In addition, the columns on which the filters are applied are displayed with a white column header as opposed to the default header color (gray).



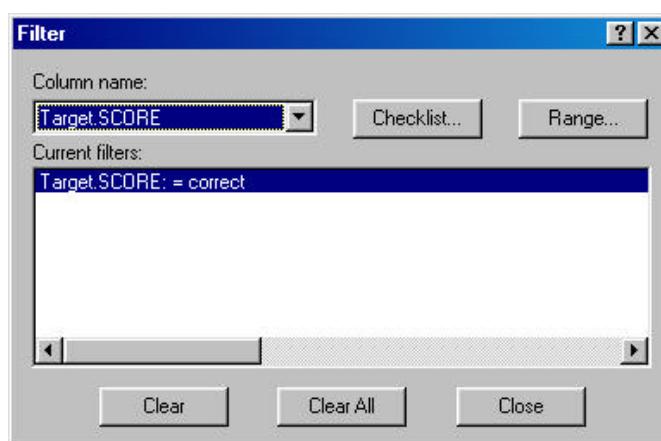
	PrimeGender	PrimeType	Procedure[Trial]	Running[Trial]	Target	Target.ACC	Target.RT	Target.SCORE	Target.SEL	TrialList	TrialList.Cycle	TrialList.Samp
1	male	negative	TrialProc	TrialList	Bob	1	211	correct	1	3	1	1
3	male	negative	TrialProc	TrialList	Linda	1	372	correct	2	4	1	3
4	female	negative	TrialProc	TrialList	Linda	1	442	correct	2	8	1	4
5	male	positive	TrialProc	TrialList	Bob	1	291	correct	1	1	1	5
6	female	negative	TrialProc	TrialList	Bob	1	308	correct	1	7	1	6
7	female	positive	TrialProc	TrialList	Linda	1	339	correct	2	6	1	7
8	female	positive	TrialProc	TrialList	Bob	1	360	correct	1	5	1	8
9	male	negative	TrialProc	TrialList	Linda	1	427	correct	2	4	1	1
11	male	positive	TrialProc	TrialList	Bob	1	388	correct	1	1	1	3
12	female	positive	TrialProc	TrialList	Bob	1	385	correct	1	5	1	4
13	female	negative	TrialProc	TrialList	Linda	1	390	correct	2	8	1	5
14	male	positive	TrialProc	TrialList	Linda	1	254	correct	2	2	1	6
15	female	positive	TrialProc	TrialList	Linda	1	275	correct	2	6	1	7
16	male	negative	TrialProc	TrialList	Bob	1	297	correct	1	3	1	8
18	female	negative	TrialProc	TrialList	Bob	1	519	correct	1	7	1	2
19	male	negative	TrialProc	TrialList	Bob	1	470	correct	1	3	1	3
20	male	positive	TrialProc	TrialList	Linda	1	338	correct	2	2	1	4
21	male	negative	TrialProc	TrialList	Linda	1	440	correct	2	4	1	5
22	female	positive	TrialProc	TrialList	Bob	1	437	correct	1	5	1	6
23	female	positive	TrialProc	TrialList	Linda	1	518	correct	2	6	1	7
24	male	positive	TrialProc	TrialList	Bob	1	377	correct	1	1	1	8
25	female	positive	TrialProc	TrialList	Linda	1	473	correct	2	6	1	1
26	female	positive	TrialProc	TrialList	Bob	1	419	correct	1	5	1	2
27	male	positive	TrialProc	TrialList	Bob	1	446	correct	1	1	1	3
28	female	negative	TrialProc	TrialList	Linda	1	319	correct	2	8	1	4
29	male	negative	TrialProc	TrialList	Linda	1	441	correct	2	4	1	5
30	female	negative	TrialProc	TrialList	Bob	1	301	correct	1	7	1	6

The status bar on the bottom right of the application display indicates the number of unhidden rows. This is the number of rows satisfying the filter criteria.

Rows Displayed: 36

5.3.8.4 Clearing Filters

Any active filters may be cleared using the Filter dialog. The Filter dialog may be displayed using the Filter command in the Tools menu, by right clicking a column header and choosing the Filter option, or by double clicking a column header.





In the Filter dialog, select the filter to be cleared and click the Clear button. To clear all filters (selected or not), click the Clear All button. Clearing filters will remove the display of the filter from the Filter dialog as well as from the Filters bar.

5.3.8.5 Refreshing Filters

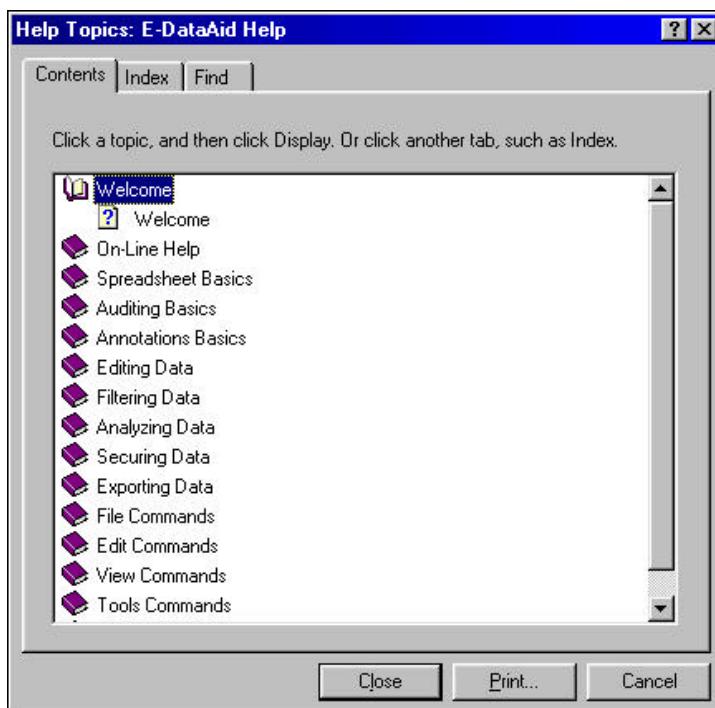
The Refresh Filters command in the View menu refreshes the filters to include edits or changes that have been made to the spreadsheet.

5.3.9 Help

The E-DataAid application supports two levels of help. The main Help system may be invoked by pressing the F1 key. The Help contents may be searched using an index listing of the available topics, or by searching for specific words or phrases in the Help topics.

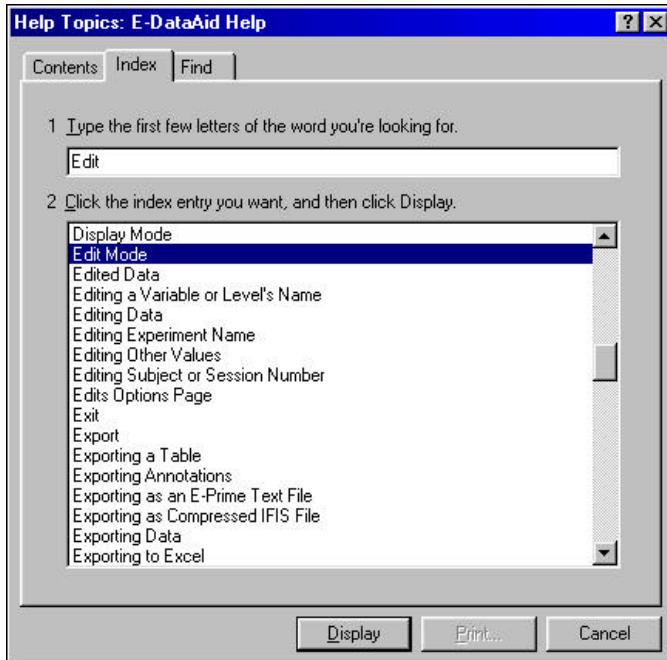
5.3.9.1 Contents

The Contents page lists a table of contents for the Help topics available within the E-DataAid Help system. The main Help topics are displayed as books, which may be opened to display related subtopics.



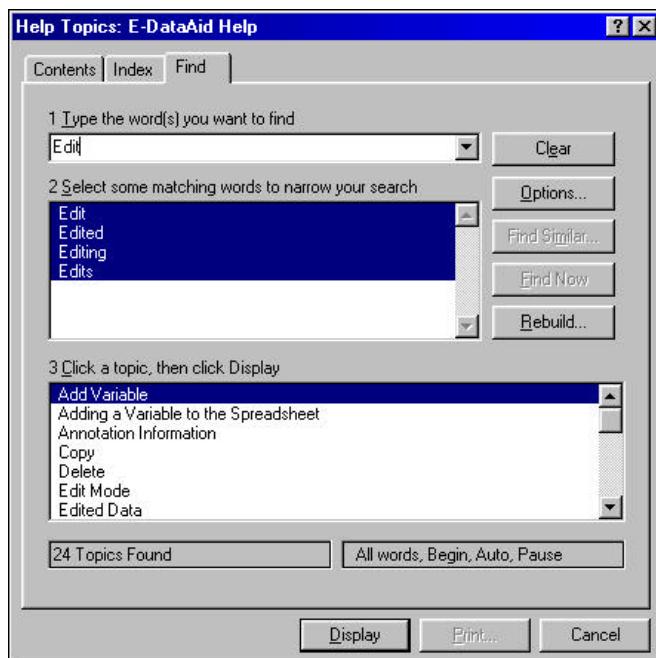
5.3.9.2 Index

The Index page displays an alphabetical listing of topics and commands within E-DataAid. The Help information for a particular topic may be displayed by selecting the topic and clicking the Display button or by double clicking the topic. The topics may be searched by typing directly in the first field of the Help Topics dialog or by scrolling through the topics contained in the index using the scroll bar on the right side of the Help Topics dialog.



5.3.9.3 Find

The Find page allows searching for specific words or phrases within the Help topics. After typing in a word to search for in the Help topics, the topics in which the word appears are listed, as well as additional suggestions to narrow the search. The Help information for a particular topic may be displayed by selecting the topic and clicking the Display button or by double clicking the topic.



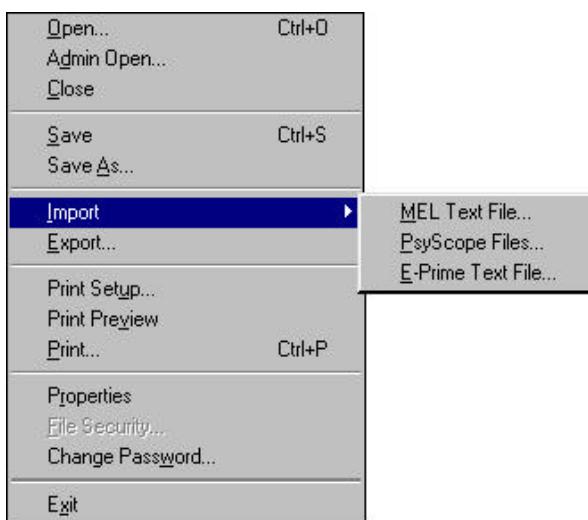


5.3.9.4 Context Sensitive Help

Context sensitive help for dialogs is available by various methods. Context sensitive help may be invoked by clicking the button in the title bar of the active dialog box and then clicking the item to be queried, or by placing the cursor over the item to be queried in the dialog box, right-clicking, and then clicking "What's This?" Finally, context sensitive help may be invoked by placing the cursor over the item to be queried in the dialog box and pressing F1.

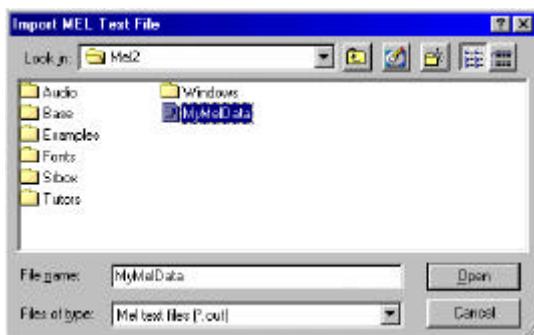
5.3.10 Importing Data Files

The Import command in the File menu may be used to import MEL Professional v2.x data saved as text files, PsyScope data files, and E-Prime data saved as text files.



5.3.10.1 MEL Professional Data File

The MEL Text file option allows the import of a MEL Professional text file containing raw data. The MEL text file must be created by exporting the MEL Professional data file (filename.DAT) to a text file using either the ANALYZE or DEDIT programs. The MEL Text File option displays the Import MEL Text File dialog box.



Navigate to the folder containing the MEL text file, select the file to import and click the Open button. The text file will be imported into E-DataAid as a new, untitled E-Prime data file (Untitled.EDAT). E-DataAid interprets all data as integers. It is not possible to import any MEL text file containing two or more sessions with the same subject and session number.

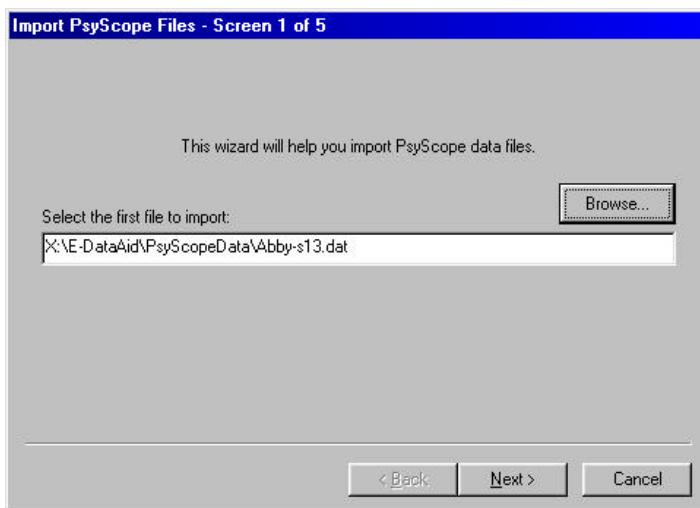


5.3.10.2 PsyScope Data File

When invoking the PsyScope Files option command to import PsyScope data files, the application displays a Wizard to aid in the import of the files. PsyScope data files are imported as new, untitled E-Prime data files. E-DataAid interprets all data as strings with the exception of subject and session numbers, which are interpreted as integers. Any PsyScope data file containing two or more sessions with the same subject and session number cannot be imported.

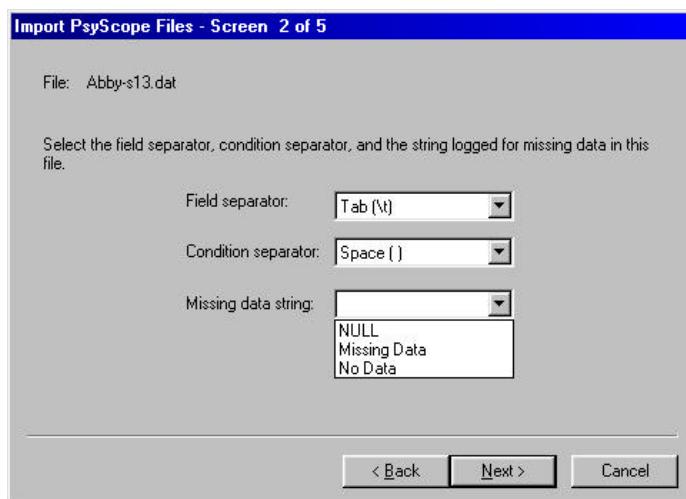
PsyScope Wizard: Screen 1

Designate the name of one of the PsyScope data files (including path) to import. This file will serve as the template for all other PsyScope data files (if any) to be imported with this file. Only files including the header information may be imported.



PsyScope Wizard: Screen 2

Select the field separator, condition separator, and the string for missing data used in the PsyScope data file.





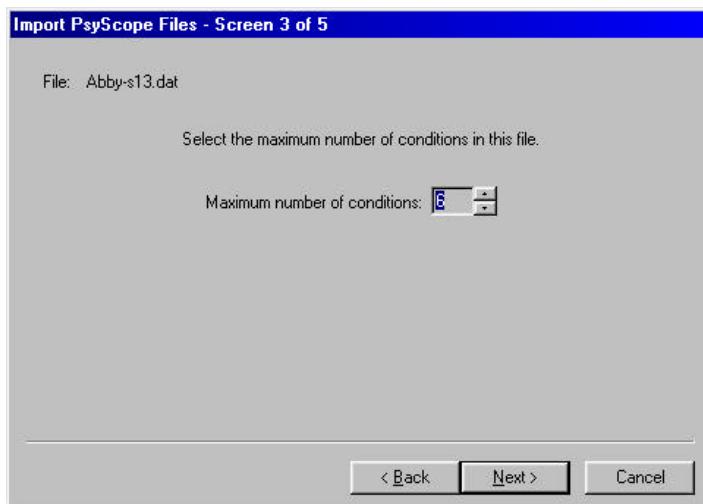
The Field separator is the value (e.g., Tab) that separates the column headings in the data section's heading line in the PsyScope data file. This separator may be a tab, space, comma, semicolon, or colon.

The Condition separator is the value that separates the conditions within the PsyScope data file's Space). This separator may be a tab, space, comma, semicolon, or colon. The "Field separator" and the "Condition separator" cannot be the same.

The Missing data string is a string E-DataAid interprets as missing data if E-DataAid encounters it in the PsyScope data files being imported. For example, if PsyScope logged "NULL" for missing data in the PsyScope data files, designate the missing data string value to be "NULL." When E-DataAid encounters "NULL", the application will flag this data as missing rather than assigning it the actual string value of "NULL".

PsyScope Wizard: Screen 3

Designate the maximum number of conditions contained in the "Condition" field in the PsyScope data file to be imported.



E-DataAid will attempt to determine this value based on the information provided in Screen 2 and the first line of data in the PsyScope data file. If the estimate is not accurate, it may be modified using the up and down arrows next to the number. If the number of conditions can vary for each trial, enter the maximum number of conditions that a trial may have. E-DataAid left-justifies the conditions matching the first condition field to Condition1, the second condition to Condition2, etc. If E-DataAid does not find the maximum number of conditions for a trial, the application flags the remaining conditions as missing data for that trial.

PsyScope Wizard: Screen 4

This screen presents a preview of the fields that the application will import. Also on this screen, select the fields in the PsyScope data file containing the subject number and session number information, and enter a string for the name of the experiment.



Import PsyScope Files - Screen 4 of 5

File: Abby-s13.dat

Preview sample of information in file:

	Variable	Value
1	PsyScope 1.1b4 started	10/23/95 10:14:17
2	Script file	NNV
3	Run on	Macintosh
4	SubjectName	abby
5	SubjectNumber	132
6	RunNumber	1
7	SubjectGroup	C
8	Age	9
9	Input devices active	BBox Key Mouse
10	Trial	3

Select the field that contains subject numbers:
SubjectNumber

Select the field that contains session numbers:
RunNumber

Type in an experiment name:
PsyExperiment

< Back Next > Cancel

PsyScope Wizard: Screen 5

Use the More Files button to select any other PsyScope data files to import with the first file selected. These files must match the first file's format and contain the same fields. Multiple PsyScope data files imported at the same time are merged to a single, untitled, E-Prime data file.

Import PsyScope Files - Screen 5 of 5

Use the More Files button to select any additional files to import with the first file. All files must have the same format as the first file. Use the Remove File button to remove a selected file from the list.

Additional files:

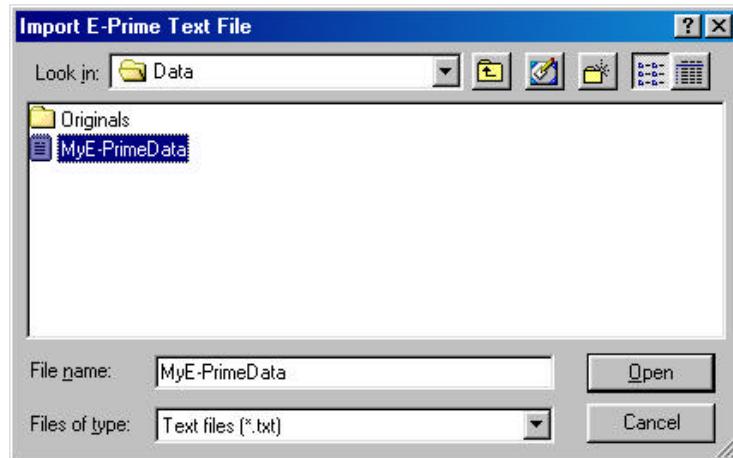
X:\E-DataAid\PsyScopeData\Abby-s13.dat

More Files... Remove File

< Back Finish Cancel

5.3.10.3 E-Prime Text File

The E-Prime Text File option allows the import of an E-Prime file containing raw data. The E-Prime text file must be created by exporting the E-Prime data file (*filename.EDAT*) to a text file using the Export command within E-DataAid. The E-Prime Text File option displays the Import E-Prime Text File dialog box.



Navigate to the folder containing the E-Prime text file, select the file to import and click the Open button. The text file will be imported into E-DataAid as a new, untitled E-Prime data file (*Untitled.EDAT*). E-DataAid interprets all data as strings with the exception of subject and session numbers, and level numbers, which are interpreted as integers.

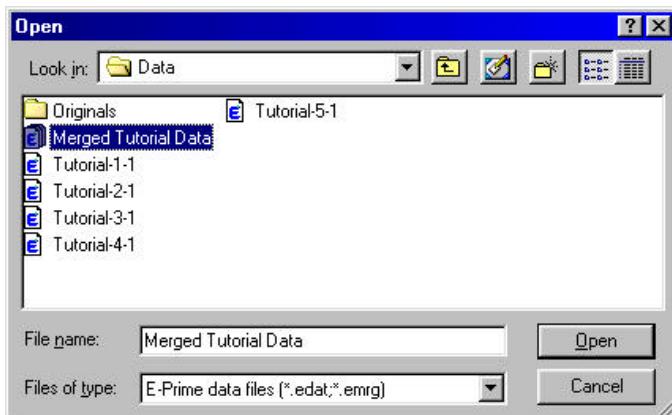
Not all E-Prime Text Files can be imported. Files which may not be imported include any files containing two or more sessions with the same subject and session number, any files not containing a subject, session or ExperimentName field, or any files containing variable fields without the corresponding level field (e.g., a file containing a field for a trial level variable, but no field for the trial level).

5.3.11 Opening a File

An E-Prime data file may be opened in E-DataAid in one of two modes. The Open command is used to open a data file as a user. The Admin Open command is used to open a data file as an Administrator, which permits access to security settings.

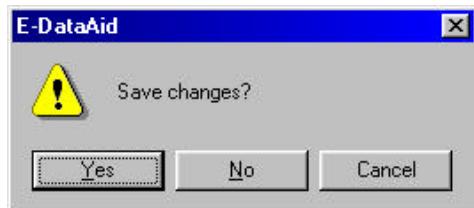
5.3.11.1 Opening as User

The Open command displays the Open dialog, which allows the user to open an existing E-Prime data file. To open a specific file, select the file or type the name in the File name field, and click the Open button.





If a file is already opened in E-DataAid and is either untitled or has been modified, E-DataAid will prompt the user to save the existing file before closing it and opening the new file. Only one file may be opened at a time.



5.3.11.2 Opening as Administrator

The Admin Open command allows the user to open a file as an Administrator, thereby having access to security options. A password is required to log in as an Administrator. The Admin Open command displays a Password dialog, requiring the user to enter a password in order to open a file.



Once a valid password is entered, the Open dialog is displayed, allowing the Administrator to select a file to open. A specific file may be opened by selecting the filename in the Open dialog and clicking the Open button (see Open). Once a file is opened by an Administrator, security restrictions for that file may be set using the File Security command on the File menu.

Password

Opening a file as an Administrator (i.e., via the Admin Open command in the File menu) requires the user to enter a password. Passwords are case insensitive and must be a string containing 4 to 8 alphanumeric characters. An encrypted version of the password is stored in the computer's registry, and is therefore machine dependent (i.e., use different Administrator passwords on different machines). Different passwords may be used to open files as an Administrator on different machines, but the same password will allow any data file to be opened as an administrator on a single machine. The first time a user opens a file in E-DataAid as an Administrator, the default password is set to "admin." After the first login, the password may be changed via the Change Password command under the File menu. The application allows three attempts to enter a valid password.

Change Password

The Change Password command displays a dialog allowing the password necessary to open a file as an Administrator to be changed. In order to change the password, the old password must be entered once, and the new password must be entered twice (i.e., to verify the new password value).



Password changes take effect immediately, when the Change Password dialog is dismissed with the OK button. Choosing Cancel dismisses the dialog without changing the password. Passwords are machine dependent, and are not saved with the data file (i.e., it is possible to have different passwords to open files as an Administrator on different machines, but the same password will allow opening any data file as an administrator on the same machine).

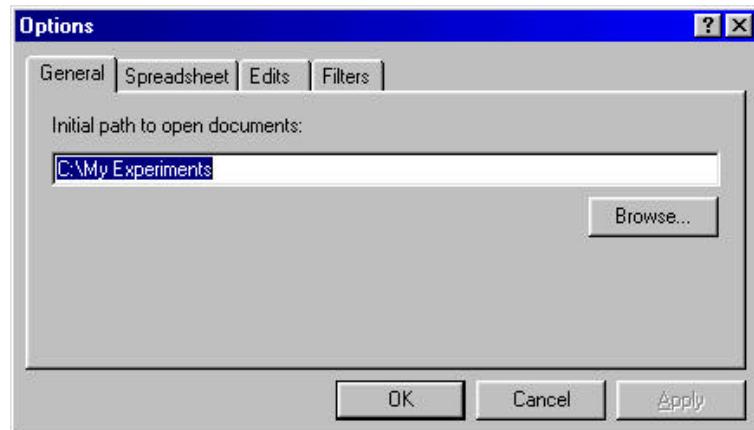
The Change Password command is active at all times. This will allow an Administrator to change the password without having to open a file.

5.3.12 Options

The Options command in the View menu allows the user to set options concerning the display of the spreadsheet. The options are machine dependent, thus the settings will be consistent across files opened on the same machine, but will not be carried to a new machine by simply copying a file. When exiting the Options dialog by selecting OK, any changes made are immediately accepted and applied to the spreadsheet. When the application is exited, the options are updated in the computer's registry.

5.3.12.1 General

General options may be examined/modified by clicking the Options command on the View menu and selecting the General tab. The General tab contains an option for the initial path to open documents.



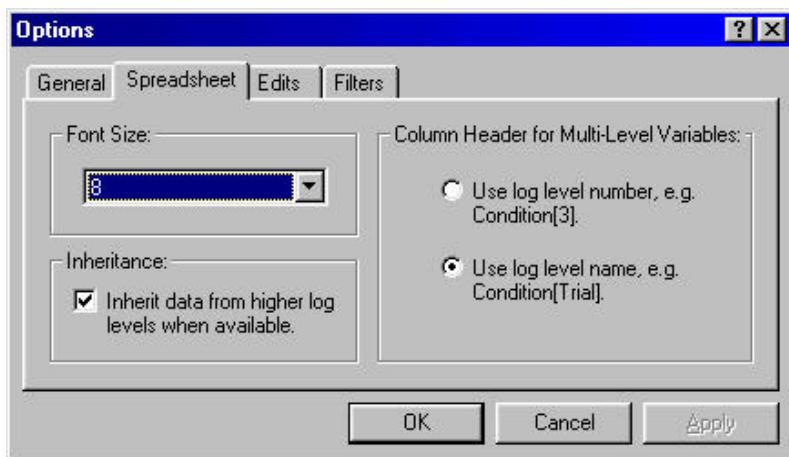


Initial Path to Open Documents

The first time the Open or Admin Open commands are used, the Open dialog will display this folder. By default, this option is set to the "C:\My Experiments" folder. This option may be changed by entering a new folder, or by using the Browse button to select a folder.

5.3.12.2 Spreadsheet

The spreadsheet options may be examined/modified by clicking the Options command on the View menu and selecting the Spreadsheet tab. The Spreadsheet tab contains options for font size, column header for multi-level variables, and inheritance.



Font Size

The Font Size is the display size of the text in the spreadsheet cells. By default, the font size is 8. To make the font size larger (10, 12, or 14), click the arrow to display the drop-down list and select the new font size.

Column Header for Multi-Level Variables

E-Prime data files allow a variable to occur at more than one level. These variables are called multi-level variables, and the spreadsheet contains a column for that variable for each level of its occurrence. For example, if the variable "Color" occurs at both the block and trial level, the spreadsheet will contain two columns for that variable: "Color[Block]" and "Color[Trial]". The first column will contain the values logged at the block level, and the second column will contain the values logged at the trial level. Column headers for multi-level variables may be displayed with the level name or the level number in brackets after the variable name. By default, the level name is used. To change this, select the level number option. For example, "Color[Block]" and "Color[Trial]" can be displayed as "Color[2]" and "Color[3]" (assuming block is level two and trial is level three for this example).

Inheritance

The Inheritance option allows the user to toggle the inheritance feature within E-DataAid on or off. By default, this feature is on. When the inheritance option is on, any multi-level variable cell that has missing data will inherit its data value from the variable's next highest level. For example, if a variable occurs at two levels (e.g., Color occurs at both the Block and Trial levels), the spreadsheet will have two columns for that variable (e.g., Color[Block] and Color[Trial]). When inheritance is on, any cells containing missing values in the lower level for the variable (e.g., Trial) will inherit the value from the next higher level (e.g., Block). Inherited values are displayed in



italics to alert the user that the value has been inherited (see Figure 1 below). When inheritance is off, the cell would contain NULL to indicate missing data (Figure 2 below).

Color[Block]	Color[Trial]
blue	<i>blue</i>
red	<i>red</i>

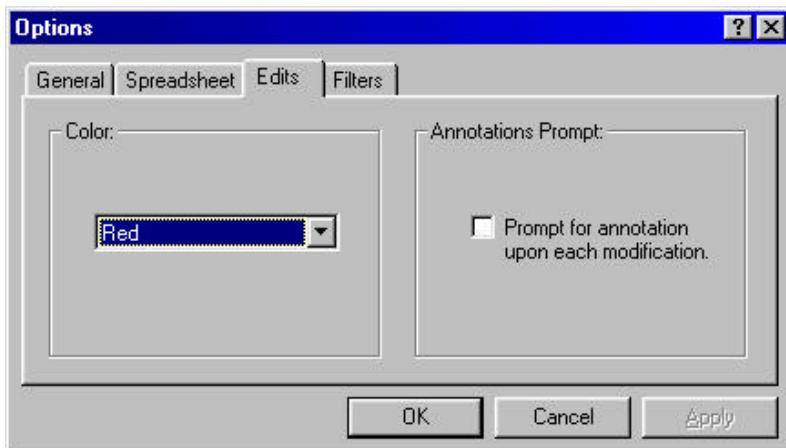
Figure 1: Inheritance turned on

Color[Block]	Color[Trial]
blue	NULL
red	NULL

Figure 2: Inheritance turned off

5.3.12.3 Edits

Edit options may be examined/modified by clicking the Options command on the View menu and selecting the Edits tab. The Edits tab contains options for color and annotation prompts.



Color

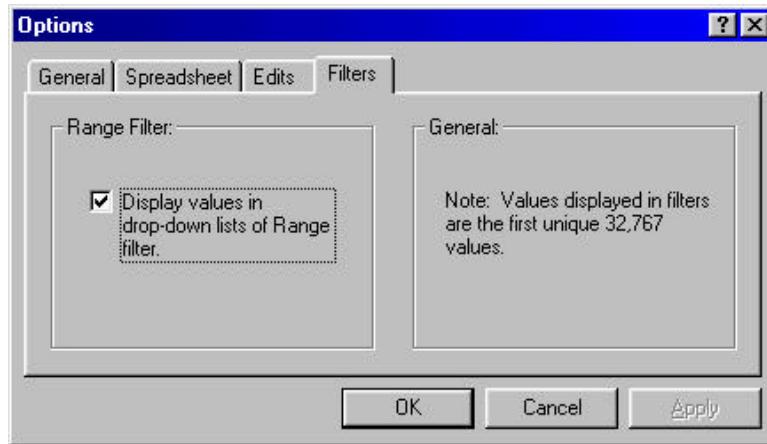
By default, the spreadsheet displays all edited values in red. To change the color of edits to either green or blue, click the arrow to display the drop-down list and select the new color.

Annotations Prompt

For all modifications to a data file, the application automatically writes an annotation to the file. An annotation records the time and date of the edit, the original value, and the modified value. User comments may be added to annotations. By default, the application does not prompt the user to enter comments when edits are made (i.e., the Annotations Prompt box is unchecked). This option may be enabled by clicking the Annotations Prompt box to check it. Comments may be added at any time using the Display Annotations command on the View menu.

5.3.12.4 Filters

Filters options may be examined/modified by clicking the Options command on the View menu and selecting the Filters tab. The Filters tab contains an option for displaying values in the drop-down lists on the Range filter.



Range Filter

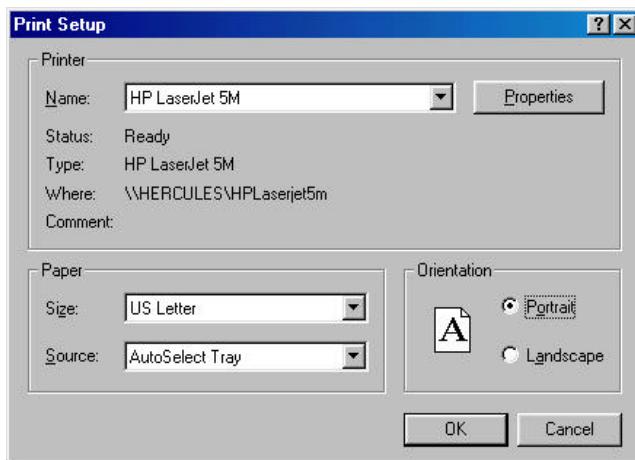
The Range filter contains fields that allow the selection of values from a drop-down list to include in the filter. By default, the Range filter displays the first 32,767 unique values of a variable. Uncheck this option to prevent the values from being displayed. In large files, because of the time required to process these unique values, it may be advisable to disable this option to reduce processing time when using the Range filter. When this option is unchecked, it is still possible to enter filter values by typing a value in the field rather than selecting it from the list. This option does not affect the checklist filter that always displays the first 32,767 unique values.

5.3.13 Printing

The Print commands in the File menu are used to set up the spreadsheet for printing, preview the print job, and send the spreadsheet to the printer.

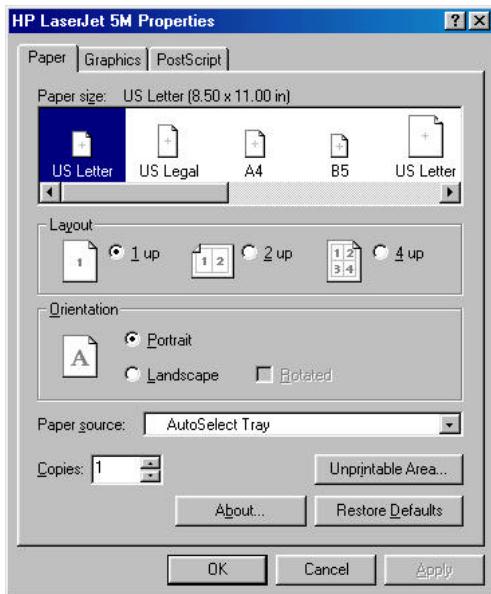
5.3.13.1 Print Setup

The Print Setup command displays the Print Setup dialog. This dialog allows the setting of the default printer, as well as the paper size and orientation. The Print Setup command allows the user to designate the print settings, but does not send the print job to the printer (see section 5.3.13.3 for additional information on printing).



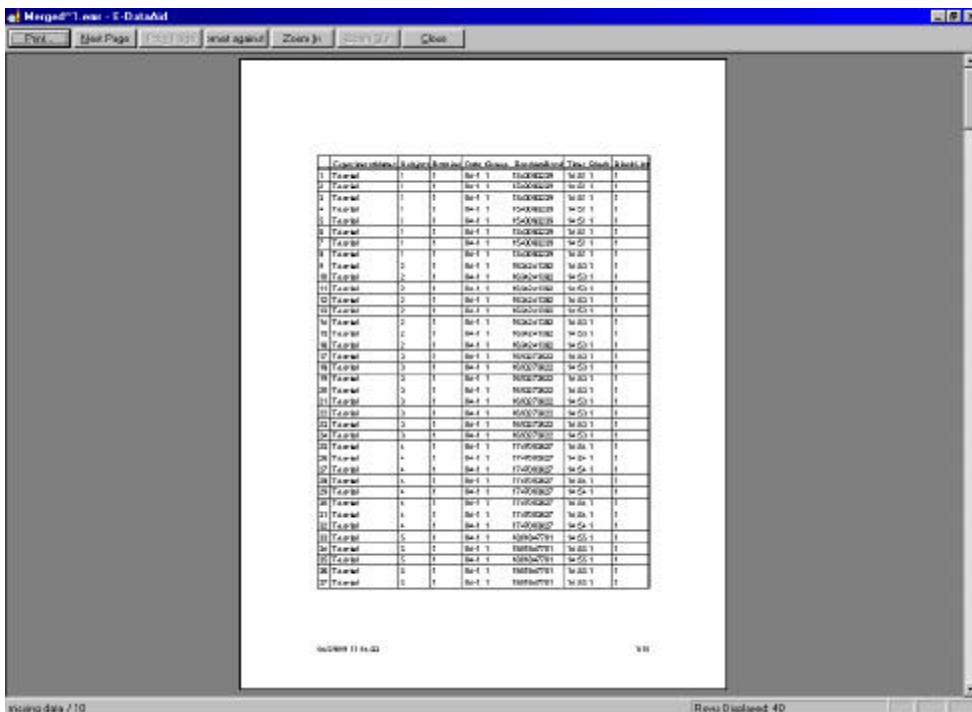


The Properties button on the Print Setup dialog allows more detailed setting of the print options.



5.3.13.2 Print Preview

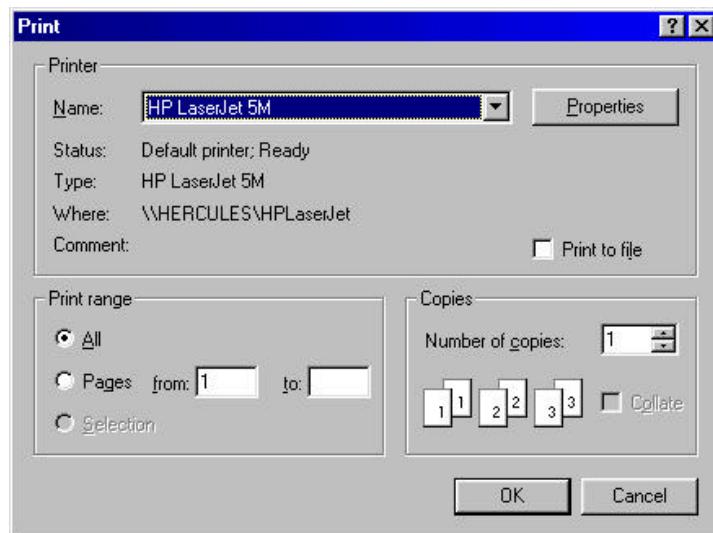
The Print Preview command displays the spreadsheet as it would appear when printed. This view allows the user to send the spreadsheet to the printer, page through the spreadsheet, view an entire page, or zoom in to view a portion of a page.





5.3.13.3 Print

The Print command displays the Print dialog, which allows the setting of the default printer, the portion of the document to print, and the number of copies.



The Properties button on the Print dialog allows for more detailed control of the print operation (see section 5.3.13.1 for Print Setup).

5.3.14 Playing a WAV file

The Play command is used to play the WAV file sound contents of a cell. The Play command may be invoked using the Tools menu, or via the Play tool button on the toolbar.



To play a WAV file, select a cell containing a WAV file value and click the Play tool button. If the cell does not contain a WAV file value, or if the WAV file cannot be located, the Play command is disabled.

5.3.15 Properties

The Properties command on the File menu allows the user to view an opened file's properties. The properties of the file provide information about the file, its contents (e.g., subject data files), and columns. This can be a quick and useful way to scan the file without scrolling through the spreadsheet.



5.3.15.1 General

The General property page may be displayed by clicking the Properties command on the File menu and selecting the General tab. The General tab displays general information about the file.



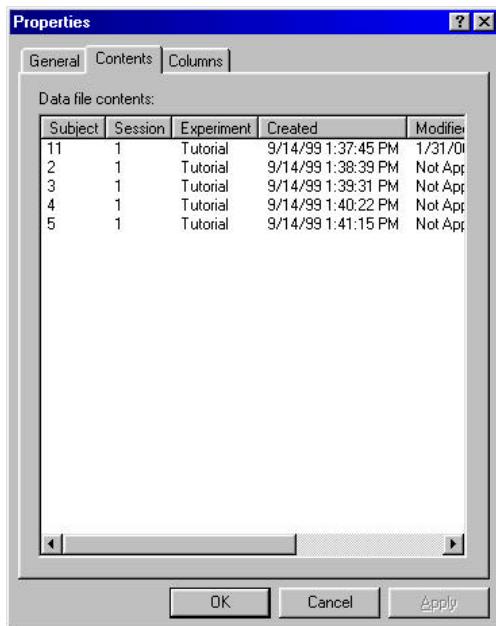
Field	Description
Filename	The filename without its path is displayed at the top of the dialog.
Type	Displays "E-Prime data file"
Version	Displays the version of the E-Prime data file format.
Status	Indicates if the data file is a "single session" file or a "merged data" file. A single session file is an E-Prime data file generated by the E-Run application. These files contain data for only one session. A merged data file is an E-Prime data file created by E-Merge or created by one of E-DataAid's import functions. These files can contain one or more sessions.
Location	Displays the file path.
Size	Displays the file size.
MS-DOS name	Displays the file MS-DOS name.
Created	Displays the file creation date.
Modified	Displays the file last modified date, which is the date that any changes were made to the file.
Merged	Displays the file last merged date, which is the date that the file was last merged into another file.
Accessed	Displays the file last accessed date.
Attributes	Displays the file attributes: Read Only, Archive, Hidden, and System. Read Only means the file cannot be written to by any user or processes. Archive means the file has been changed since the last backup. Hidden means the file is hidden and does not appear in normal directory searches. System means that the file is a system file and is excluded from normal directory searches.

Additionally, if the file has altered data, the property page will include the message "This file has data alterations!" below the Size field. A file contains altered data if a data value or a variable name has previously been modified.



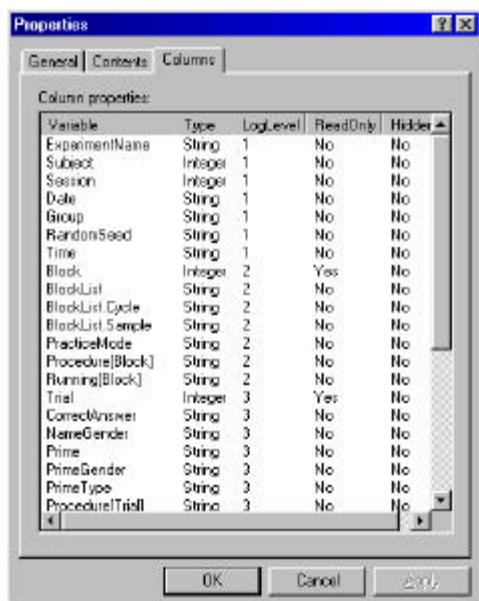
5.3.15.2 Contents

The Contents property page may be displayed by clicking the Properties command on the File menu and selecting the Contents tab. The Contents tab displays a list of all sessions contained in the file along with their creation date, last modified date, and last merged date.



5.3.15.3 Columns

The Columns page may be displayed by clicking the Properties command on the File menu and selecting the Columns tab. This tab lists the column name, data type, level, read-only status, and hidden status. Columns are listed in the order in which they appear in the spreadsheet.





5.3.16 Saving a Data File

The Save command in the File menu saves the opened data file to the same name. If no name has been specified, the Save As dialog will be displayed, allowing the user to enter the name of a file to which the data will be saved.

5.3.17 Security

The File Security command in the File menu permits a user who opens a file as an Administrator to set restrictions for that file concerning which variables may be viewed or edited. When a file is opened by a User (using the Open command), the User is subject to the security restrictions set on that file by an Administrator, and a User does not have access to the File Security command or settings (i.e., the File Security command in the File menu is disabled).

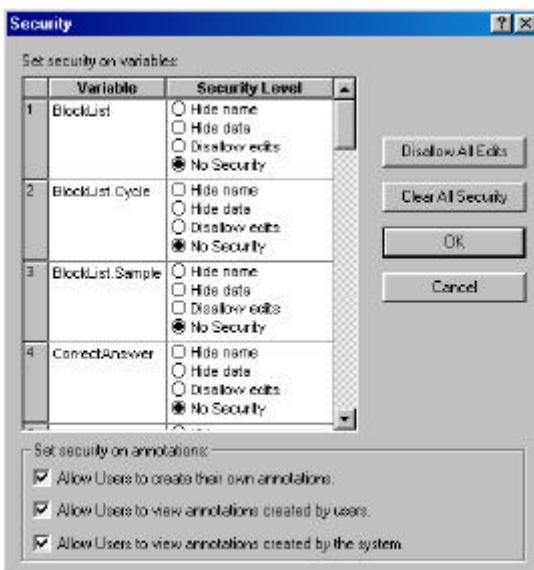
5.3.17.1 Opening a File as an Administrator

To set security restrictions, the user must open a file as an Administrator via the Admin Open command in the File menu, and must supply a password (see section 5.3.11.2 for Password). Administrators are not subject to any security restrictions aside from the password required to open a file as an Administrator. Therefore, when setting security restrictions, the restrictions will not be seen in the spreadsheet until the application is closed and reopened as a User.

Security restrictions are saved with the file. The Administrator must save the file prior to exiting E-DataAid, or the restrictions will be lost. Since the restrictions are saved with the file, the restrictions will apply on any machine to which the file is moved. Administrator passwords are not saved with the file, but are machine dependent. Different passwords may be used on different machines (see section 5.3.11.2 for Password).

5.3.17.2 Variable settings

The File Security command displays the Security dialog, allowing the Administrator to individually set security levels for each variable in the file. One option for the security setting may be chosen from the four available security options. Security for a multi-level variables may be set only once, even though it appears in more than one column in the spreadsheet.





Hide Name

The Hide Name option is the highest level of security. Hide Name will display no column for a variable with the Hide Name setting in the spreadsheet. When a file containing a variable with the Hide Name restriction is opened by a User (i.e., using the Open command), there will be no indication of that variable in the spreadsheet, and the User will not be able to unhide the column.

Hide Data

The Hide Data option will display a column for the variable in the spreadsheet, but will not display any data for the variable. When a file containing a variable with the Hide Data restriction is opened by a User (i.e., using the Open command), the variable name will be displayed in the column header, but the column will be grayed, and no data will be displayed in the cells in the column (i.e., they will remain blank).

Disallow Edits

The Disallow Edits option will display a column for the variable in the spreadsheet and the data for the variable will be displayed, but the data will be read-only. When a file containing a variable with the Disallow Edits option is opened by a User (i.e., using the Open command), the variable name and data for that variable will be displayed, but the User will be unable to edit the name of the variable, or any of the data cells for that variable. If all edits are disallowed, E-DataAid will also not allow the creation of new variables, or the editing of log-level names (see section 5.3.17.3 for Disallow All Edits).

No Security

The No Security option will display a column for all variables in the spreadsheet, the data for all variables is displayed, and all data may be edited. By default, all of the variables within a data file are set to the No Security option, which is the lowest level of security.

5.3.17.3 General Settings

Clear All Security

The Clear All Security button removes all security restrictions from all variables without having to apply this setting individually to each variable. This button also removes all security restrictions on annotations (i.e., checks all three boxes related to Annotations security).

Disallow All Edits

The Disallow All Edits button sets the security option for all variables to Disallow Edits without having to individually apply this setting to each variable. This button also unchecks the "Allow

5.3.17.4 Annotation Settings

Security for annotations may be set generally. E-DataAid does not support the setting of security options for individual annotations. If an Administrator is preventing Users from viewing a variable or its data, it may be prudent to also restrict the User from viewing any system or user-created annotations (i.e., the annotations may contain references to the hidden data).



Set security on annotations:

<input checked="" type="checkbox"/> Allow Users to create their own annotations.
<input checked="" type="checkbox"/> Allow Users to view annotations created by users.
<input checked="" type="checkbox"/> Allow Users to view annotations created by the system.

Create User Annotations

By default, Users may add their own annotations to a data file using the Create New button on the Annotations dialog. The Annotations dialog may be displayed using the Display Annotations command in the View menu. The creation of user annotations may be restricted by an Administrator. If the option to allow users to create annotations is unchecked in the annotations security settings, a User (i.e., using the Open command to open a file) may not create new annotations or add text to an existing annotation. If this option is unchecked, system annotations are still automatically created for all file modifications. Security options for allowing the creation of user annotations may be set by an Administrator using the File Security command in the File menu.

View User Annotations

User annotations are created by displaying the Annotations dialog via the Display Annotations command in the View menu, and choosing the Create New button on the Annotations dialog. By default, Users may view all user annotations. The viewing of annotations may be restricted by an Administrator. If the option to allow the viewing of annotations created by users is unchecked in the annotations security settings, a User (i.e., using the Open command to open a file) may not view user-created annotations. Security options for viewing user annotations may be set by an Administrator using the File Security command in the File menu.

View System Annotations

By default, Users may view all annotations created by the system. System annotations are the annotations the application automatically logs for all edits. The viewing of annotations may be restricted by an Administrator. If the option to allow users to view annotations created by the system is unchecked in the annotations security settings, Users (i.e., using the Open command to open a file) may not view system annotations. Security options for viewing system annotations can be set by an Administrator using the File Security command in the File menu.



Chapter 6: E-Recovery

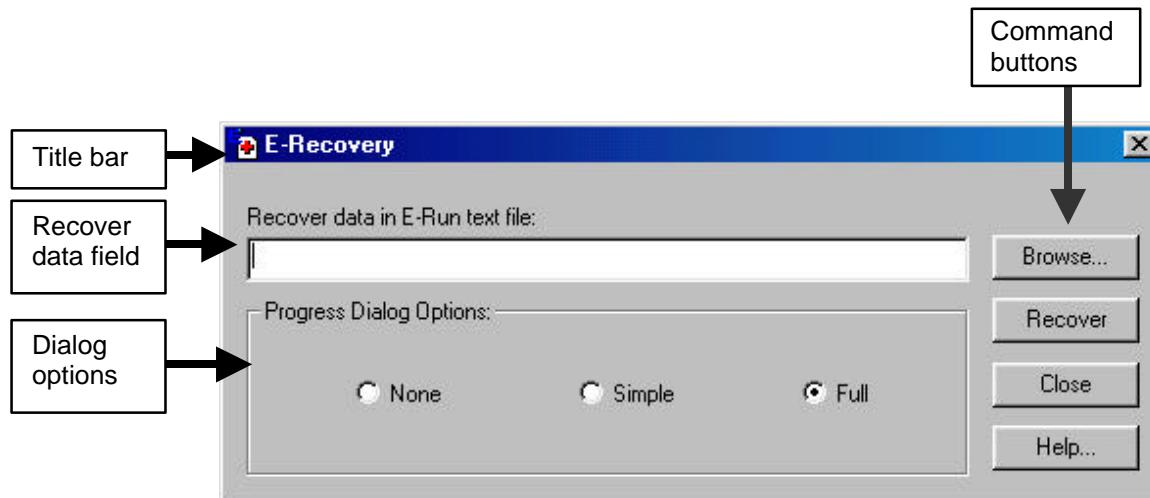
6.1 About E-Recovery

The E-Run application logs data to a text file as an experiment is in progress. The data in this text file is a flat text file. When the experiment is completed, the E-Run application converts this flat file into an E-Prime data file. The data in the E-Prime data file is in a Spreadsheet format, which enables the hierarchical structure of the data to be displayed. E-Run logs data to the text file in a reversed arrangement, so the deepest levels are logged foremost trailed by their parent levels. For example, in an experiment with two blocks of ten trials in each block, E-Run would log the data for the first ten trials followed by the data for the first block. Then the data for the second ten trials would be logged, followed by the data for the second block, and then finally data for the session. By using E-DataAid, the E-Prime data file will display the data in a usable and comprehensible spreadsheet format.

However, there may be times when the conversion from the E-Run text file (TXT) to the E-Prime data file (EDAT) does not occur (e.g., the experimenter terminates an experiment early by pressing Ctrl+Alt+Shift), or the EDAT file may simply be missing. The E-Recovery program converts a TXT file selected by the user into an EDAT file. The E-Run application logs a limited amount of recovery information to the beginning of each E-Run text file. This information will include at least the experiment name, the subject number, and the session number. E-Recovery will use this information when reconstructing the session data.

6.2 About the Interface

The E-Recovery interface is comprised of four command buttons: Browse, Recover, Close and Help (See Section 6.3 for information about launching the E-Recovery application). The title bar is displayed at the top of the application screen, and displays the E-Recovery icon and the name of the application (i.e., E-Recovery). The “Recover data in E-Run text file” field will display the text file that is about to be converted to an E-Prime data file.





6.2.1 The Progress Dialog Options

The radio buttons in the “Progress Dialog Options” give choices for three levels of reporting the progress of the recovery procedure. The dialog is only displayed when the TXT file is of considerable length and requires a substantial amount of time to convert to an EDAT file.

Option	Description
None	Provides no information concerning the recovery of the data file.
Simple	Provides partial information concerning the recovery of the data file.
Full	Provides complete information concerning the recovery of the data file.

6.2.1.1 Full

The Full option displays a dialog box providing complete information concerning the recovery of the data file. It rapidly exhibits each block and line as it recovers, in addition to higher-level information (i.e., Subject #, Session #, etc.).



6.2.1.2 Simple

The Simple option displays a dialog box providing partial information concerning the recovery of the data file. It rapidly exhibits each line number as the file is recovered.





6.2.1.3 None

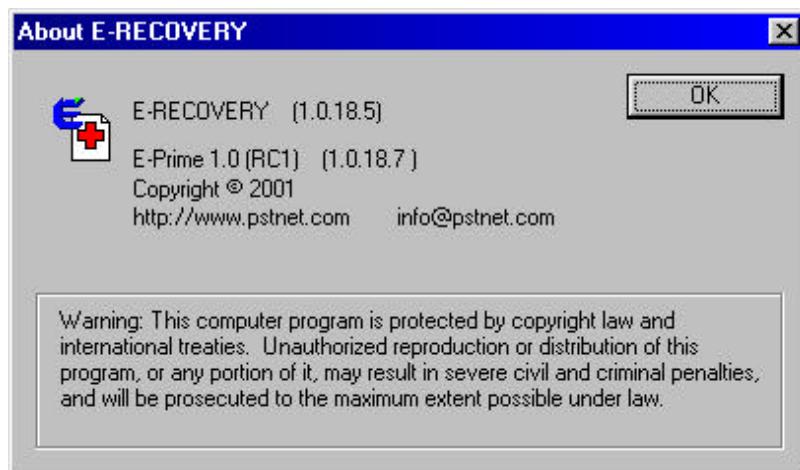
The None option provides the choice of not seeing any dialog box as the data file is recovered. The hourglass icon will be displayed as the data file is recovered.

6.2.2 Command Buttons

Commands	Shortcut	Function
Browse		Selects the E-Run text file (*.TXT) to be converted to an EDAT extension.
Recover		Converts the data in the text file into an E-Prime data file.
Close	ALT+F4	Closes the E-Recovery program.
Help		Provides version information about the E-Recovery program.

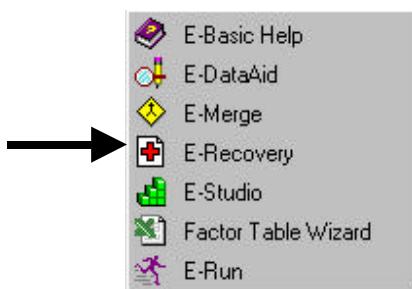
6.2.2.1 About E-Recovery

To display application information, right click the E-Recovery icon in the upper left corner of the E-Recovery dialog and select the About E-Recovery command. The About E-Recovery dialog displays version information as well as contact information for Psychology Software Tools, Inc.



6.3 Using E-Recovery

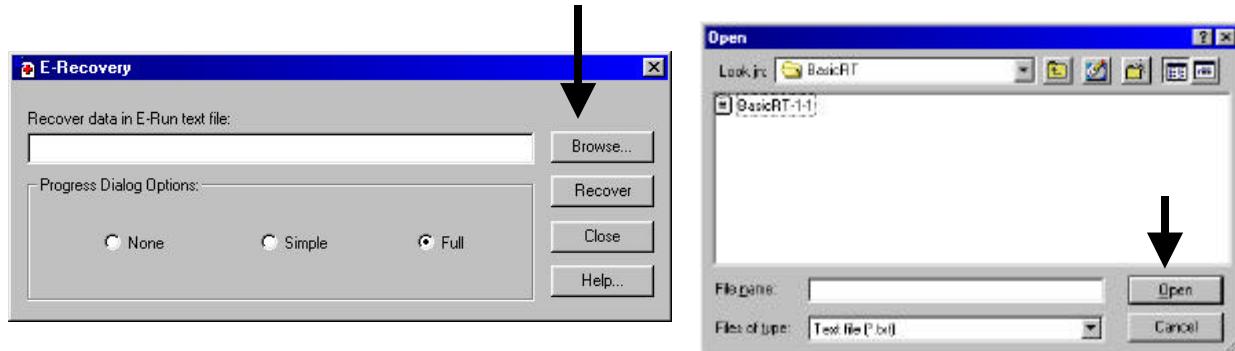
E-Recovery is launched through the E-Prime menu from the Start button. From the Start button, choose Programs and the E-Prime menu. From the E-Prime menu, select E-Recovery.



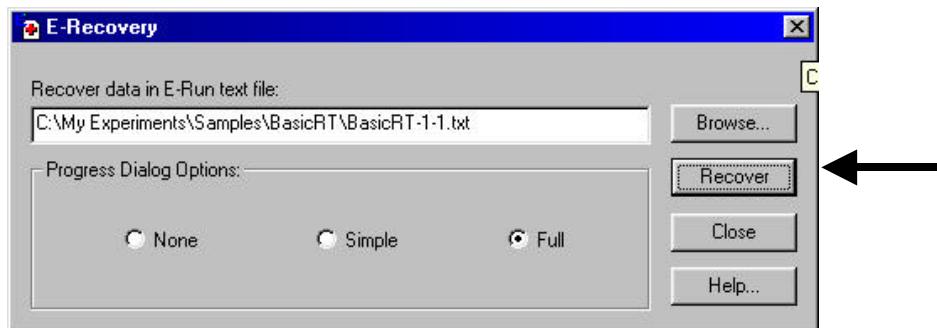


6.3.1 Recovering Files

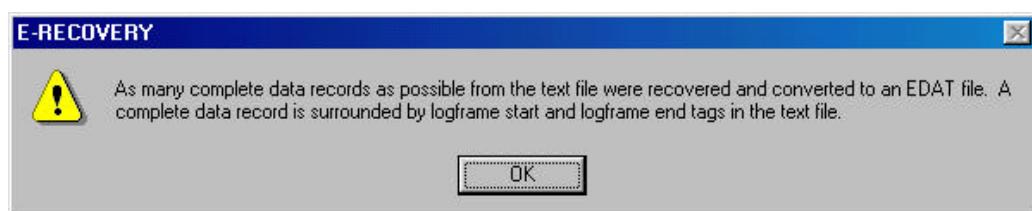
Select the E-Prime TXT file to be converted to an EDAT file by clicking the Browse button located within the E-Recovery dialog. Navigate to the folder containing the TXT file, select the file, and click Open.



Once the file is selected, the name of the text file will appear in the edit control. Use the Recover button to convert the data in the text file (TXT extension) into an E-Prime data file (EDAT extension).



After clicking the Recover button a dialog box will appear, explaining that as many data records as possible were recovered. Click "OK" to complete the procedure.

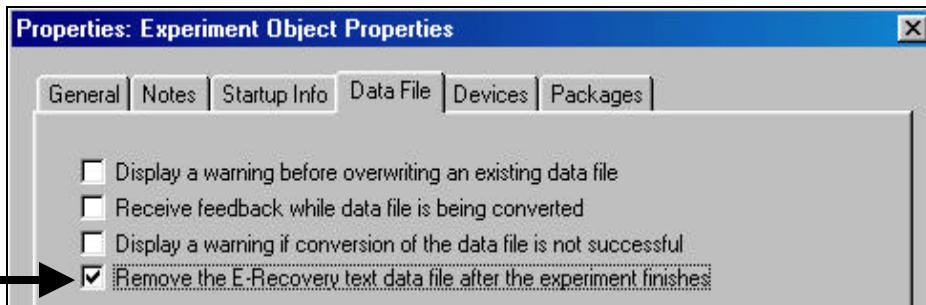


6.3.2 Viewing Recovered Data

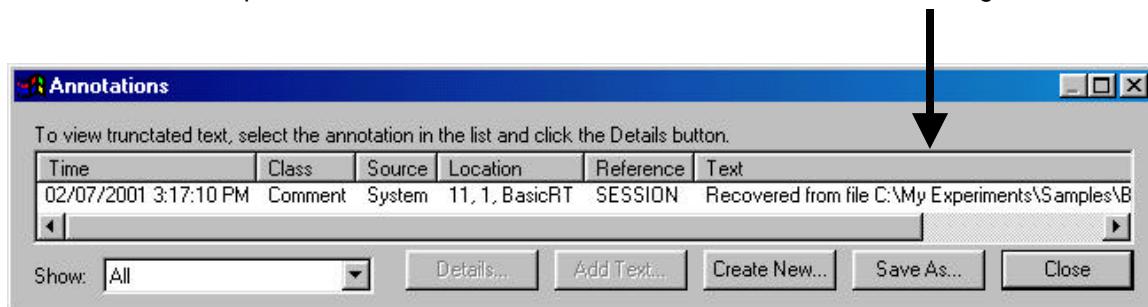
The converted E-Prime data file will have the same name as the E-Run text file with the exception of the extension. The converted file receives the EDAT extension. For example, if the E-Run text file is called "BasicRT-11-1.txt", the recovered E-Prime data file will be called "BasicRT-11-1.edat". Once the E-Prime data file is recovered, the data may be viewed using E-DataAid. Refer to Chapter 5 - E-DataAid in the Reference Guide for further information about E-DataAid and opening files in E-DataAid.



E-Studio includes an optional setting to delete the TXT file after successful completion of the data file conversion to an EDAT file. In E-Studio, open the Experiment object's Property pages from the Structure view. Select the Data File tab, and check the box preceding "Remove the E-Recovery text data file after the experiment finishes". Click OK to accept the new setting and dismiss the dialog. By default, this option is not activated. For further information about the Experiment object, refer to section 1.3.2 of Chapter 1 - *E-Studio* in the Reference Guide.



E-Prime data files converted by E-Recovery will have an annotation in E-DataAid that indicates the session was recovered from an E-Run text file. This practice distinguishes E-Prime data files created using the E-Recovery program from files created initially by the E-Run application. Refer section 5.3.2 of Chapter 5 in the Reference Guide for further information concerning annotations.



6.3.3 Troubleshooting – Common Data Conversion Problems

In the E-Run application, when the TXT file does not successfully convert to an EDAT file a dialog box will appear at the completion of the experiment. It instructs the use of the E-Recovery utility to recover data.





There are several common causes for a data file not to convert and several solutions discussed below.

6.3.3.1 Data File Open

A conversion error will occur if an experiment is running with certain startup information (i.e., Subject, Session, etc.), and a data file with the same information is already open in E-DataAid. The data file may have been opened for viewing and not closed prior to running again. E-Run will not convert a TXT file to overwrite an EDAT file if the EDAT file is already in use. To rectify, close the data file in E-DataAid. The current run was saved as a TXT file, and can be successfully converted to an EDAT file by using the E-Recovery program once the EDAT file is closed.

6.3.3.2 Reserved Words

A reserved word might have been used in the user-written script within E-Studio. Reserved words, or keywords, are words recognized by E-Basic as part of the E-Basic language, and cannot be used to create a variable, function, constant, or subroutine. The solution to this error is to simply avoid using any reserved words in the user-written script. However, all keywords can be used for names of structure members. For a complete list of all reserved words, refer to E-Basic Help, and type "keywords" or "reserved words" in the Find tab's search field.

For example, if an attribute is created on a TrialList called "Subject", it is referencing a variable that is pre-defined globally by E-Prime, hence can only exist at the Session level. When it occurs at any lower level, a logging conflict occurs, and a data file conversion error results. The data logging will log correctly if the attribute is renamed to a unique identifier in the TrialList and/or the TextDisplay object that presents it.

6.3.3.3 Logging Turned Off

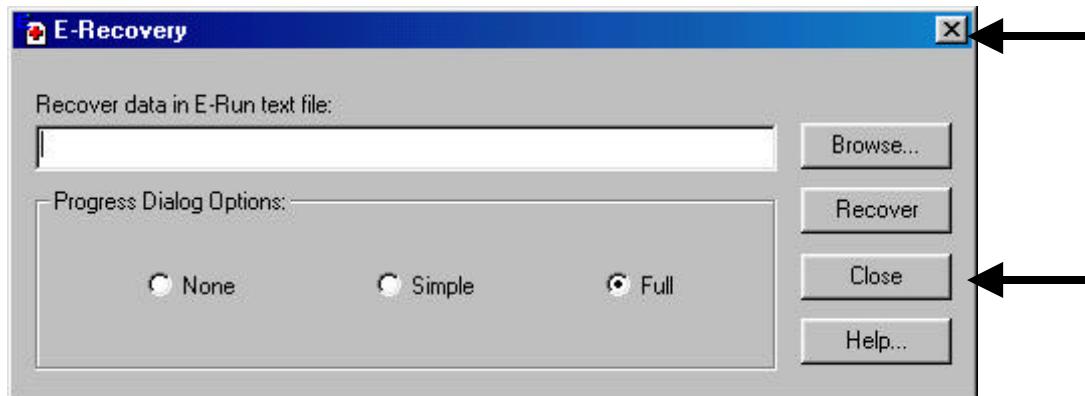
The Property for logging data (i.e., LogData) for a Procedure could have been set to "No" on the SessionProc and/or TrialProc objects. It is possible to shut off data logging for any level except at the Session Level. Without logging the Session level though, the critical variables of the ExperimentName, Subject, and Session will not be logged. Set LogData to "Yes" for the Session Level by selecting the SessionProc in the experiment Structure, and setting the LogData property via the Properties window.

6.3.3.4 Extra Tab After an Attribute

When calling an attribute through user-written script, an extra tab may have been added after c.GetAttrib resulting in a run-time error. Thus, E-Basic reads it as printing the number of spaces necessary to achieve a given column position. To correct the error, simply delete the extra tab that was placed after the attribute.

6.3.4 Exiting E-Recovery

To exit E-Recovery, select the Close button, click on the "X" (i.e., in the upper right hand corner of the dialog box), or press Alt+F4.



6.3.5 Getting Help

E-Recovery's interface presents a Help button for getting assistance and searching for items. Click on the Help button to display the on-line help for E-Recovery.



**E-Prime Reference Guide
Chapter 6: E-Recovery**

This page intentionally left blank.



Appendix A - Error List

Error#	Displayed Message	Error Description	Solution
10000	The allowable response was unrecognized: <i><response></i> <i><response> = string value that was not recognized</i>	E-Basic: The Allowable parameter in the InputDevice.CreateInputMask is not recognized by the input mask. E-Studio: The value entered in the Allowable field in the Response Options (Duration/Input Property Page) is not recognized by the input mask.	Enter device specific values for allowable response.
10001	The allowable response cannot be empty	E:Basic: The Allowable parameter for InputDevice.CreateInputMask must contain a value that is recognized by the input mask. E-Studio: The Allowable field in Response Options (Duration/Input Property Page) must contain a value when input is enabled.	Enter device specific values for allowable response.
10002	Cannot have duplicate response	E-Basic: The Allowable parameter in InputDevice.CreateInputMask includes duplicate response entries. E-Studio: A single value has been entered more than once. Since the Allowable field in the Response Options (Duration/Input Property Page) represents a mask of values, entering duplicate values would be redundant.	Verify that unique entries are entered only once.
10004	Experiment terminated by user	The CTRL+ALT+SHIFT keypress combination was detected, signaling the E-Run runtime engine to terminate.	n/a
10005	Experiment abnormally terminated	An error occurred in the runtime that caused the experiment to terminate abnormally.	Contact technical support if this error can be replicated on a regular basis.
10006	Assertion Failed	An error was thrown due to a Debug.Assert statement evaluating to False. It is a good programming practice to place numerous Debug.Assert statements throughout an experiment to ensure it is operating to design specifications.	Evaluate experiment specification to determine cause for the Debug.Assert statement evaluating to False.
10007	Cannot call .Run when DeviceManger is suspended	A call to RteRunnableObject.Run is not permitted while the DeviceManager is suspended. For more information, refer to the DeviceManager.Suspend command in the E-Basic Online Help.	Call DeviceManager.Resume to permit E-Basic runtime operations to be permitted.
10008	Port value out of range	The value for RteRunnableInputObject.OnsetSignalPort or RteRunnableInputObject.OffsetSignalPort is invalid.	Enter valid port address values for this property.



E-Prime User's Guide
Appendix A: Error List

Error#	Displayed Message	Error Description	Solution
10010	Invalid Context	The Context parameter passed to Procedure.Run is not valid.	Specify a valid parameter of Context data type to the function.
10011	Cannot call Suspend when already Suspended	Cannot call DeviceManager.Suspend when the DeviceManager is already suspended.	Call DeviceManager.Resume to resume the DeviceManager before suspending a subsequent time.
10012	Cannot call Resume without first calling Suspend	Cannot call DeviceManager.Resume without a call to DeviceManager.Suspend to suspend the DeviceManager.	Call DeviceManager.Suspend first, before resuming the DeviceManager with a call to DeviceManager.Resume.
10013	CreateInputMask not allowed when device is emulating	When a device is opened to emulate its input responses into another device, it does not process input for itself. Therefore, a call to InputDevice.CreateInputMask is not permitted.	Either stop the device from emulating its input responses to another device, or create the input mask on the device into which it is emulating.
10014	MaxCount specified is too large	The value specified for the maximum number of entries for the collection exceeds the largest value the collection can hold.	Specify a value less than or equal to 4096, the largest value a collection can hold.
10015	Invalid EchoClients Parameter	The value specified for the EchoClients parameter for InputDevice.CreateInputMask is not valid. The EchoClients parameter must be a variable of RteObject data type and hold a collection of EchoClient objects.	Specify a value for the EchoClient parameter that represents a collection of valid EchoClient objects.
10016	Invalid EndResponseAction	The value specified for the EndResponseAction parameter for InputDevice.CreateInputMask is invalid.	Specify a value that is valid for the EndResponseAction parameter. The constants, ebEndResponseActionNone, ebEndResponseActionTerminate, and ebEndResponseActionJump can be used for these values.
10017	Invalid Termination Response	E-Basic: The value specified for the TerminationResponse parameter for InputDevice.CreateInputMask is not valid or recognized by the input mask. E-Studio: The value entered in the Termination Response field in the Advanced Response Options (Duration/Input Property Page) is not recognized by the input mask.	Enter device specific values for the termination response.
10018	Correct Response is not part of Allowable	E-Basic: The value entered in the Correct parameter for the InputDevice.CreateInputMask is not included as part of the Allowable Response parameter. E-Studio: The value entered in the Correct Response field in the Response Options (Duration/Input Property Page) is not included as part of the Allowable Response Options.	Specify a value that is included as part of the allowable response.



Error#	Displayed Message	Error Description	Solution
10019	Termination Response is not part of Allowable	E-Basic: The value entered in the TerminationResponse parameter for the InputDevice.CreateInputMask is not included as part of the Allowable response parameter. E-Studio: The value entered in the Termination Response field in the Advanced Response Options (Duration/Input Property Page) is not included as part of the Allowable response Options.	Specify a value that is included as part of the allowable response.
10020	Unknown Custom Option: '<options>' <options> = string representation of the custom options not recognized by the input mask	The value entered in the CustomOptions parameter for the InputDevice.CreateInputMask is not recognized by the input mask.	Specify device specific custom options for this parameter.
10022	Cannot terminate a mask that is not armed.	An error was thrown as a result of a call to InputMask.Terminate at a point where the input mask was not armed to accept input responses. An input mask is automatically armed when the RteRunnableInputObject it belongs to .Run is called. An input mask can become unarmed when it times out, accepts max count responses, termination response occurs, or is terminated in response to another mask being terminated.	Call InputMask.Terminate only when the InputMask is armed to accept input responses.
10023	Invalid Echo Client Found	An error was thrown due to the EchoClients parameter for InputDevice.CreateInputMask containing an object that is not of the EchoClient data type.	Call InputDevice.CreateInputMask with a EchoClients parameter that represents a collection of objects that are of data type EchoClient.
10024	Mask does not support reserved object	An error was thrown due to the ReservedObject parameter for InputDevice.CreateInputMask being specified for a mask that does not recognize, support, or require an extended object to be specified.	Call InputDevice.CreateInputMask with a ReservedObject parameter only for input masks that require a reserved object.
10025	Unavailable Owner	An error was thrown due to an object of type InputMask not being part of a RteRunnableInputObject input mask collection. (e.g., after a trial completes and the running object no longer requires the input mask.)	Obtain a reference to an input mask prior to an object's run and only use until the end of the trial.
10026	Cannot specify both before and after index	An attempt to add an object will fail with a runtime error when both the Before and After parameters are specified for the RteCollection.Item call. The Before and After parameters are mutually exclusive and only one should be specified.	Call RteCollectionAdd with Before or After specified, but not both.



E-Prime User's Guide
Appendix A: Error List

Error#	Displayed Message	Error Description	Solution
10027	Invalid after index	The After parameter for RteCollection.Add must represent a valid integer index or string key into the collection. An error will be thrown when the After parameter represents any other data type.	Call RteCollection.Add with a value for After as an integer index or string key into the collection.
10028	Invalid before index	The Before parameter for RteCollection.Add must represent a valid integer index or string key into the collection. An error will be thrown when the Before parameter represents any other data type.	Call RteCollection.Add with a value for Before as an integer index or string key into the collection.
10029	Cannot add null to a collection	The RteCollection data type holds objects that are of RteObject or derived data type. An error will be thrown if the value for the Item parameter for RteCollection.Add represents any other data type.	Call RteCollection.Add with a value for the Item parameter that represents a valid object of type RteObject.
10030	Add functionality for this collection is not supported.	The specific collection does not permit items to be added to it as a result of a call to RteCollection.Add.	n/a
10031	Remove functionality for this collection is not supported.	The specific collection does not permit items to be removed from it as a result of a call to RteCollection.Remove.	n/a
10032	RemoveAll functionality for this collection is not supported	The specific collection does not permit all items to be removed from it as a result of a call to RteCollection.RemoveAll	n/a
10033	This key is already associated with an element of this collection	The Key parameter uniquely identifies a mechanism to recall an object from a collection. An error will occur if an attempt is made to add an item with a Key parameter matching that of an existing item in the collection. The string key will remain until the object is removed from the collection.	Call RteCollection.Add with a unique string key.
10034	Invalid before and after index	An invalid value was specified for the Before and After parameters for RteCollection.Add. The Before or After parameter for RteCollection.Add must represent a valid integer index or string key into the collection.	Call RteCollection.Add with a value for Before or After as an integer index or string key into the collection.
10035	Unable to add item to collection	The call to RteCollection.Add failed due to the specific collection not supporting the data type attempting to be added.	Call RteCollection.Add with an Item parameter containing a valid value for the collection.
10036	Unable to remove item from collection	The call to RteCollection.Remove failed due to the Index parameter not being valid.	Call RteCollection.Remove with an Index parameter representing a valid value for the collection.



Error#	Displayed Message	Error Description	Solution
10037	Internal Error: Cannot find before key	The call to RteCollection.Add failed due to the Before parameter specifying an integer index or string key that is not valid for the collection. The numeric index starts at 1 and must be less than or equal to RteCollection.Count.	Call RteCollection.Add with a Before parameter that is valid for the collection.
10038	Internal Error: Cannot find after key	The call to RteCollection.Add failed due to the After parameter specifying an integer index or string key that is not valid for the collection. The numeric index starts at 1 and must be less than or equal to RteCollection.Count.	Call RteCollection.Add with a After parameter that is valid for the collection.
10039	The value for the KEY parameter is either invalid or cannot be found in this collection	An operation on RteCollection failed due to the Key parameter not specifying a valid string key of an item in the collection.	Call RteCollection methods with valid string key entries of items in the collection.
10041	The value for the INDEX parameter is either invalid or is not within range for this collection	An operation on RteCollection failed due to the Index parameter not specifying a valid integer index of an item in the collection. The numeric index starts at 1 and must be less than or equal to RteCollection.Count.	Call RteCollection methods with a valid index to entries of items in the collection.
10042	Name can only be set once	The RteObject.Name property may be written only once. Once the RteObject.Name property is set, it cannot be renamed.	n/a
10043	Name cannot be empty	E-Basic: The RteObject.Name property cannot be an empty string. E-Studio: An object was named in the interface with no value. An object name can only contain A-Z, a-z, 0-9, must begin with an alphabetic character, must not contain more than 80 characters, and must be unique within the system.	E-Basic: Specify a valid string value for the RteObject.Name property E-Studio: Enter a valid name through the Properties window, Procedure time line, or Structure view windows.
10044	The name cannot be longer than 80 characters!	E-Basic: The RteObject.Name property cannot have more than 80 characters. E-Studio: An object was named in the interface with a length of more than 80 characters. An object name can only contain A-Z, a-z, 0-9, must begin with an alphabetic character, must not contain more than 80 characters, and must be unique within the system.	E-Basic: Specify a valid string value for the RteObject.Name property that does not have more than 80 characters E-Studio: Enter a valid name through the Properties window, Procedure time line, or Structure view windows that does not have more than 80 characters



E-Prime User's Guide
Appendix A: Error List

Error#	Displayed Message	Error Description	Solution
10045	Name must begin with a letter.	<p>E-Basic: The RteObject.Name property must begin with an alphabetic character.</p> <p>E-Studio: An object was named in the interface with a value that did not start with an alphabetic letter. E-Object names are generated to E-Basic script, which does not permit variable names to begin with a numeric value.</p> <p>An object name can only contain A-Z, a-z, 0-9, must begin with an alphabetic character, must not contain more than 80 characters, and must be unique within the system.</p>	<p>E-Basic: Specify a valid string value for the RteObject.Name property that starts with an alphabetic character</p> <p>E-Studio: Enter a valid name through the Properties window, Procedure timeline, or Structure view windows that does starts with an alphabetic character</p>
10046	Name contains an invalid character.	<p>E-Basic: The RteObject.Name property cannot include any invalid characters.</p> <p>E-Studio: An object was named in the interface with a value that contains an invalid character. E-Object names are generated to E-Basic script, which does not permit variable names to have invalid characters.</p> <p>An object name can only contain A-Z, a-z, 0-9, must begin with an alphabetic character, must not contain more than 80 characters, and must be unique within the system.</p>	<p>E-Basic: Specify a valid string value for the RteObject.Name property that does not include any invalid characters.</p> <p>E-Studio: Enter a valid name through the Properties window, Procedure time line, or Structure view windows that does not have any invalid characters.</p>
10048	A parameter passed to the method/function is invalid	Methods and Properties require the parameters passed to them to be of a certain data type. An error will occur any time a parameter is passed to a method or property that does not represent the data type expected.	Specify parameters to methods and properties that are of the correct data type.
10049	The InputMask cannot be added to this InputMaskManager because it already has been previously added to another InputMaskManager	A call to InputMaskManager.Add can only accept input mask objects that are not already part of another input mask manager.	Only add input mask objects to one InputMaskManager object.
10050	The version is read only	The specific values returned from the Version object are read-only and cannot be modified.	n/a
10051	An error occurred while attempting to open the device: <error> <error> = hexadecimal representation of the error code returned	The device could not be opened due to the specific error. Typically this error occurs as result of a hardware or driver issue. The device must be connected to the system and the proper driver must be installed prior to the run of the experiment. A device may not support the configuration specified for it in the structure parameter for opening that device.	<p>E-Basic: Verify that the device is connected, has a proper driver, and that it supports the configuration in the structure passed to the Device.Open call.</p> <p>E-Studio: Verify the device is connected, has a proper driver, and that it supports the configuration in the experiment specification.</p>



Error#	Displayed Message	Error Description	Solution
10052	Emulation Device is invalid or not open. The order in which the devices load may need to be changed	When a device is set to emulate, the device into which input is to be emulated must be opened first.	E-Basic: Organize calls to Device.Open so that the device being emulated into calls Device.Open first. E-Studio: Position the order of the devices in the Experiment Object property pages; Devices tab with the Move Up and Move Down buttons.
10053	The device must be open prior to calling this method	Methods and properties of objects of the Device data type must have a successful call to Device.Open prior to being called.	Ensure that Device.Open is called prior to calling methods/properties on objects of the Device data type.
10054	The operation invoked is currently not supported	The method or property called is not supported by the version of E-Prime, operating system used, or license agreement of the installed software.	Review the documentation and license agreement to ensure that the property/method can be called in the installed versions of E-Prime and the operating system.
11000	Cannot set source color key: <error> <error> = hexadecimal representation of the error code returned	Video cards or drivers that do not support source color key operations will cause the error to be thrown during a call to set Canvas.SourceColorKey.	Verify that the video card hardware supports source color key operations.
11001	Invalid Source Canvas	The value for the SourceCanvas parameter for Canvas.Copy was not of data type Canvas.	Call Canvas.Copy with a SourceCanvas parameter of data type Canvas.
11006	Invalid Source Rect	The value for the SourceRect parameter for Canvas.Copy was not valid. SourceRect requires a variable of data type Rect and must have coordinates that are within the rectangle of the display device.	Call Canvas.Copy with a SourceRect parameter that represents a Rect data type with coordinates within the rectangle of the display device.
11007	Invalid Destination Rect	The value for the DestRect parameter for Canvas.Copy was not valid. The DestRect parameter requires a variable of data type Rect and must have coordinates that are within the rectangle of the display device.	Call Canvas.Copy with a DestRect parameter that represents a Rect data type with coordinates within the rectangle of the display device.
11008	SaveImage Failed	A call to Canvas.SaveImage failed. This may be due to a bad file name, a write-protected disk, or the file already being open.	Determine that the filename exists, the file is closed, and that the disk is not write-protected.
11016	Cannot load bitmap file <file> <error> <file> = string representation of the file that could not be loaded. <error> = string representation of further error information	A call to Canvas.LoadImage failed. This may be due to a bad file name, the file already being open, or specifying a file that is not supported by the Canvas object. Windows DIB (device independent bitmap) files with a BMP extension optionally with RLE compression are supported.	Call Canvas.LoadImage with a Filename parameter representing a string data type that points to a valid path for a Windows device-independent bitmap image.



E-Prime User's Guide
Appendix A: Error List

Error#	Displayed Message	Error Description	Solution
11017	Invalid ActiveState	A call to set Slide.ActiveState failed due to the SlideState not being valid.	Ensure the ActiveState being set is a valid string name and that the SlideState has been properly added to the SlideState collection.
11020	Collection only supports SlideState objects	A call to SlideStateCollection.Add failed due to the Item parameter representing an object that was not of the SlideState data type.	Call SlideStateCollection.Add with an Item parameter that represents an object of the SlideState data type.
11021	Collection only supports up to ten SlideState objects	A call to SlideStateCollection.Add failed due to the maximum number of SlideState objects being exceeded.	Call SlideStateCollection.Add no more than the maximum number of SlideState objects the collection can hold.
11022	Collection only supports SlideStim objects	A call to SlideStimCollection.Add failed due to the Item parameter representing an object that was not of the SlideStim data type.	Call SlideStimCollection.Add with a Item parameter that represents an object of the SlideStim data type.
16000	Invalid Address	E-Basic: The value for PortDevice.Port is invalid. E-Studio: The value for Port device Port field is invalid in the Experiment object property pages; Devices tab for the Port device.	Enter valid port address values for this property.
16001	Invalid Size	E-Basic: The value for PortDevice.Port is invalid. E-Studio: The value for the Port device Size field is invalid in the Experiment object property pages; Devices tab for the Port device.	Enter valid size values for this property.
17000	Invalid Port Number	E-Basic: The value for SerialDevice.Port is invalid. E-Studio: The value for the Serial device Port field is invalid in the Experiment object property pages; Devices tab for the Serial device.	Enter valid port address values for this property.
18000	Error setting Pan Value: <error> <error> = hexadecimal representation of the error code returned	Unable to set the Pan value for the sound object. The valid range of values is between -10000 and 10000. Panning features are not supported by all sound hardware.	Ensure that the value specified for the Pan parameter is within the range of -10000 to 10000 and that the hardware supports the setting of pan values.
18001	Error Setting Volume Value: <error> <error> = hexadecimal representation of the error code returned	Unable to set the Volume value for the sound object. The valid range of values is between -10000 and 0. Volume features are not supported by all sound hardware.	Ensure the value specified for the Volume parameter is within the range of -10000 to 0 and that the hardware supports setting of volume values.



Error#	Displayed Message	Error Description	Solution
18002	Collection only supports SoundBuffer objects	A call to SoundBufferCollection.Add failed due to the Item parameter representing an object that was not of the SoundBuffer data type.	Call SoundCollection.Add with a Item parameter that represents an object of the SoundBuffer data type
18004	Cannot create sound buffer: <error> <error> = string representation of the error message returned	An error occurred while attempting to create an object of the SoundBuffer data type. This can be due to lack of resources, invalid configuration, or improper driver.	Ensure that the configuration of the file matches the configuration of the sound device.
18005	Cannot load sound file <file>. <error> <file> = string representation of the file that could not be loaded. <error> = string representation of further error information	An error occurred while attempting to load a sound file. This can be due to not being able to find the file, the file is already open, lack of resources, invalid configuration, or improper driver.	Ensure that the configuration of the file matches the configuration of the sound device, that the filename and path are correct, and that the file is not already open.
18006	Invalid Active Buffer	The object to which the ActiveBuffer property in the SoundBufferCollection points does not represent an object with a SoundBuffer data type.	Ensure that the ActiveBuffer property is set to a valid object of the SoundBuffer data type
18007	Invalid Buffer Index	The value being set to the ActiveBuffer property is not within valid range for the collection. The start index is 1 and the maximum value is equal to the SoundBufferCollection.Count value.	Ensure that the value of the ActiveBuffer is between 1 and SoundBufferCollection.Count.
18008	Cannot set the current position unless the sound buffer is playing	A call to set SoundBuffer.CurrentPosition can only occur while the SoundBuffer is currently playing.	Call SoundBuffer.Play prior to setting the value of SoundBuffer.CurrentPosition.
18009	The value for CurrentPosition must be between start and stop offset	A call to set SoundBuffer.CurrentPosition must be a valid integer value between the values of SoundBuffer.StartOffset and SoundBuffer.StopOffset.	Call to set SoundBuffer.CurrentPosition must be a valid integer value between the values of SoundBuffer.StartOffset and SoundBuffer.StopOffset.
18010	The value for MaxLength is invalid	E-Basic: The MaxLength parameter of the SoundBufferInfo structure represents a value that is not valid. Valid values for SoundBufferInfo.MaxLength cannot be less than 100. E-Studio: The MaxLength field of the SoundOut object (General property page) represents a value that is not valid. Valid values for MaxLength cannot be less than 100.	E-Basic: Set the value of SoundBufferInfo.MaxLength to a value no less than 100 E-Studio: Set the value of MaxLength to a value no less than 100



E-Prime User's Guide
Appendix A: Error List

Error#	Displayed Message	Error Description	Solution
18011	The value for BitsPerSample is invalid	<p>E-Basic: The BitsPerSample parameter of the SoundBufferInfo structure represents a value that is not valid. Common values are 8 and 16, but values are hardware dependent.</p> <p>E-Studio: The BitsPerSample field of the Sound device (Experiment object; Devices tab) represents a value that is not valid. Common values are 8 and 16, but values are hardware dependent.</p>	<p>E-Basic: Set the value of SoundBufferInfo.BitsPerSample to 8 or 16.</p> <p>E-Studio: Set the value of BitsPerSample to 8 or 16.</p>
18012	The value for Channels is invalid	<p>E-Basic: The Channels parameter of the SoundBufferInfo structure represents a value that is not valid. Valid values are 1 for mono and 2 for stereo.</p> <p>E-Studio: The Channels field of the Sound device (Experiment Object; Devices tab) represents a value that is not valid. Valid values are 1 for mono and 2 for stereo.</p>	<p>E-Basic: Set the value of SoundBufferInfo.Channels to 1 or 2.</p> <p>E-Studio: Set the value of Channels to 1 or 2.</p>
18013	The value for SamplesPerSecond is invalid	<p>E-Basic: The SamplesPerSecond parameter of the SoundBufferInfo structure represents a value that is not valid. Common values are 11025, 22050, 44100, but values are hardware dependent.</p> <p>E-Studio: The SamplesPerSecond field of the Sound device (Experiment Object; Devices tab) represents a value that is not valid. Common values are 11025, 22050, 44100, but values are hardware dependent.</p>	<p>E-Basic: Set the value of SoundBufferInfo.SamplesPerSecond to 11025, 22050, or 44100</p> <p>E-Studio: Set the value of SamplesPerSecond to 11025, 22050, or 44100</p>
19001	Invalid Port Number	<p>E-Basic: The Port field of the SRBoxDeviceInfo structure represents a value that is not valid. Common values are 1, 2, 3, and 4, but are system specific.</p> <p>E-Studio: The Port field of the SRBox device (Experiment Object; Devices tab) represents a value that is not valid. Common values are 1, 2, 3, and 4, but are system specific.</p>	<p>E-Basic: Set the value of SRBoxDeviceInfo.Port to 1, 2, 3, or 4.</p> <p>E-Studio: Set the value of Port to 1, 2, 3, or 4</p>
19002	Invalid Lamp Mode	A call to CLampMode failed due to an invalid mode being specified. Valid modes are "Normal", "Toggle", and "Sticky".	Set the value passed into CLampMode to "Normal", "Toggle", or "Sticky".
19005	Invalid Voice Key Trip Level Value	A call to SRBoxDevice.VoiceKeyTripLevel failed due to an invalid setting. Valid values for the voice key trip level are between 0 and 31.	Set the value of SRBoxDevice.VoiceKeyTripLevel between 0 and 31.
19006	Invalid Key Bank Value	A call to set SRBoxDevice.KeyBank failed due to an invalid setting. Valid key bank values are 0 and 1.	Set the value of SRBoxDevice.KeyBank to 0 or 1.
19007	Invalid Lamp Bank Value	A call to set SRBoxDevice.LampBank failed due to an invalid setting. Valid key bank values are 0 and 1.	Set the value of SRBoxDevice.LampBank to 0 or 1.

INDEX

A

- abstract.....32
Administrator.....*See* opening a file
Advanced Device properties
 Collection.....25
 Echo26
Allow Collapsing option.....11
Analysis154–67
 Batch.....166
 Binning data.....157–58
 closing.....166
 collecting multiple statistics.....156–57
 creating155–57, 166
 loading166
 running.....158–65
 saving.....165–66
ANL.....*See* Analysis
Annotations.....168–72
 creating170–71
 displaying.....168–69
 exporting to text.....171–72
 security.....211–12
 text169–70
 viewing Details171
attributes14
 sorting14
audio formats*See* WAV files
auxiliary files111

B

- Batch Analyses166–67
 adding167
 copying/plotting to Excel.....167
 exporting.....167
 loading167
 removing167
 reordering.....167
 saving.....167
Binning data.....157–58
bitmaps (BMP).....73, 111
bracket/attribute notation . 61. *See also* TextDisplay object
Browser.....14–15
 Cut, Copy, and Paste objects.....15
 delete objects15
 sorting objects.....15

C

- Canvas object.....93
cells
 columns.....175–77
 collapse levels176–77
 hide/unhide175–76
 restore order176
 editing152, 180
 Inheritance option152

- missing data.....153
navigation.....153
 NULL value.....153
collapsing11, 149, 176
compile103, 107
Context menus.....1, 8, 16–17
converting TXT to EDAT format...*See* E-Recovery
copy protection.....106

D

- data alteration168
data file37
 conversion*See* Experiment object
 Log Level names38
 properties207–9
Debug object93
Devices
 adding23, 39
 advanced properties24
 editing40
 multiple response input24
 Properties40
 removing24, 39
dragging and dropping icons18
Duration/Input22
 input devices23
 input mask22
 Jump Label22

E

- ebs file103–6
 color defaults106
EDAT file116, 119, 126, 134, 135
E-DataAid
 analyzing154
 Annotations168
 closing177
 editing178
 exiting185
 exporting185
 filtering189
 Help194–96
 importing196
 opening200–202
 options202–5
 playing a WAV file207
 Printing205–7
 Properties207–9
 saving210
 Spreadsheet172
edit data178–84
 cell180
 copying182
 delete182
 experiment name179
 Fill Down.....182
 find183

paste	182
renaming variables.....	181–82
replace.....	184
subject and session number.....	179
undo	183
variables.....	180–81
E-Merge	
annotations.....	121
conflicts	122–25, 133, 141
filtering	125–26
icons.....	118, 129
merging files	129–34
opening a data file.....	141
Quick Reference	144
recursive merge.....	133
shortcuts.....	140
standard merge.....	131
EMRG file	116, 119, 126, 130, 134, 135
E-Objects	<i>See also</i> specific objects
FeedbackDisplay object.....	76
ImageDisplay object	60
InLine object.....	83–84
Label object	89
List object	45
PackageCall object.....	90–93
Procedure object	43
Scripting objects	
Canvas object.....	93
Debug object.....	93
Summation object	93
Slide object	65
SoundOut object	84
TextDisplay object.....	54
Wait object.....	88–89
E-Recovery	
data conversion problems	217–18
Progress Dialog Options	214–15
Full.....	214
None.....	215
Simple.....	214
recovering data files.....	216
view recovered data	216–17
errors.....	107–9
compile	107
error list.....	A-1
run-time	107
E-Run.....	55, 61, 103–13, 217
function shortcuts.....	105
E-Studio	1
Experiment object	42–43. <i>See also</i> E-Objects
author and abstract information	32
Data File	35–38
Log Level Names.....	38
packages.....	42–43
parameters.....	32–37
Startup Info	42–43
subject information	32
exporting data	185–89
to E-Prime Text.....	187–88
to Excel.....	185–86
to other formats	188–89
to SPSS.....	186
to StatView.....	186–87
F	
FeedbackDisplay object	78. <i>See also</i> E-Objects
accuracy.....	79–80
format	80
input.....	79
macros	82
open	76
reaction time	79–80
selection of objects	78
sub-objects.....	76
File List view.....	139
fields	119
filtering	125–26
icons	118, 129
filtering	189–94
Checklist filter.....	189–90
clearing	193–94
display	192–93
Range filter.....	190–92
refreshing.....	194
Folder Tree view	117
Folder Tree View	139
Format	80
Frame.....	19
alignment	20
border	21
G	
generate	103, 107
global properties	31
screen resolution.....	31
sound properties	31
H	
hardware key	106–7
Help	126–28, 194–96
Contents.....	127, 194
Context Sensitive.....	196
Find	128, 195
Index.....	127, 194
I	
ImageDisplay object.....	60–64. <i>See also</i> E-Objects
open	60
pictures	61–62
properties	62–63
importing data files.....	196–200
E-Prime text	199–200
MEL Professional.....	196
PsyScope	197–99
InLine object	83–84. <i>See also</i> E-Objects
Color Defaults	84
properties	84
User written script	83

input mask	
echo.....	26–28
multiple masks	22–24
J	
Jumping	18, 22, 87, 88–89
L	
Label object	89. <i>See also</i> E-Objects
Jumping	89
properties	89
List object	45–53. <i>See also</i> E-Objects
attributes	48, 52
controls and features	48
create.....	46
delete attributes.....	47
delete levels	47
fields	51
hide levels.....	52
Load Method	
Embedded	49
logging attributes	<i>See</i> Logging
open	46
order of levels	50
Counterbalance	50
Offset	50
Permutation.....	50
Random.....	50
Random With Replacement	50
Sequential	50
properties	46
run-time sample and cycle sizes 51. <i>See also</i> List	
object fields	
spreadsheet grid	47–48
summary	47
Listed Parameters.....	<i>See</i> Startup Info
Load Method.....	49
Log Level Names	
editing	<i>See</i> data file
rules for naming	38
Logging.....	29, 52
attributes	52
Startup Info	32
status	52
M	
max count	<i>See</i> string responses
menus	
Context.....	1, 150
E-DataAid.....	148–50
E-Merge	148–50
E-Run.....	148–50
E-Studio	148–50
Shortcut.....	1
Merge Input History	137, 143
Merge Log View.....	120, 140
merge operations.....	130. <i>See also</i> merging files
Recursive Merge.....	133–34
Standard Merge.....	131–33
Undo Merge.....	136
Merge Output History	138, 144
Merged data files	115, 119, 126, 143
merging files.....	129–34. <i>See also</i> E-Merge
Merge Input History	137
merge operations	130
Merge Output History	138
merge results	135–36
security options	139
source file	129
Target file	130
Undo Merge.....	136
N	
nested.....	44
Notes	31
O	
Objects.....	<i>See also</i> E-Objects
copy	12
Cut, Copy and Paste	15
delete	11, 15
move	12
open	12
rename	11
sort.....	15
opening a file.....	200–202
Administrator	201–2
User	200–201
options	141, 202–5
conflicts	141
Edits.....	204
Filters.....	204
General	202
Initial Display Folder.....	141
Spreadsheet.....	203–4
outline of an experiment.....	10
Output window	17
Debug	17
Generate	17
P	
PackageCall object	90–93. <i>See also</i> E-Objects
install	90–91
open	92
properties	92–93
parent objects.....	11
collapse	11
expand	11
plotting	163–64
port communications	23, 39–41
Printing	205–7
procedural timeline	18, 43, 84, 89
Procedure object	43–45. <i>See also</i> E-Objects
log data	45
nested.....	44
sequence of events.....	43
Progress Dialog Options.....	<i>See</i> E-Recovery
Properties.....	<i>See</i> E-Objects

using attributes.....	13
Properties window	12–13
editing	13
Property Pages	19–31
PsyScope.....	<i>See</i> importing data files
R	
reaction time	79–80
recovering data files.....	<i>See</i> E-Recovery
Recursive Merge operation	133–34
S	
screen resolution	31
Script window	15, 1–16, 106
color defaults	106
Full	15
syntax highlighting	16
Toggle Bookmark	16
User.....	15
Scripting objects	93. <i>See also</i> E-Objects
access	93
Canvas object.....	93
Debug object.....	93
Summation object	93
Security	
open as Administrator.....	201, 210
options	139, 210–12
Password.....	201–2
session level variables.....	32
Shortcut menus	<i>See</i> Context menus
shortcut to experiment	111–12
Single Session files	119, 129
Slide object	64–75. <i>See also</i> E-Objects
create.....	65
properties	67–68
SlideState	68
sub-object.....	69
SlideState object	68–71
background	70
enable and clean-up	70
properties	71
SlideImage sub-object.....	73
SlideSoundOut sub-object	75
SlideText sub-object	72
SlideStim object.....	<i>See</i> sub-objects
sound properties.....	31
SoundOut object	84–87. <i>See also</i> E-Objects
audio file properties	86
buffer	84
open	84
source file 121–25, 129, 131–33, 135–37, 139, 141	
spider down.....	11
Spreadsheet	
arranging.....	172
cells.....	152–53
columns.....	173–76
Filters Bar.....	154
Status Bar	154
SPSS	<i>See</i> exporting data
Standard Merge operation	131–33
Startup Info.....	32–35
choice data type	36
fields	33
Parameters	32
adding	34
editing	35
Listed.....	34
removing	34
User-Specified	34
subject information.....	32
Status	119
Merged data.....	119
Single Session	119
Target file	119
StatView	<i>See</i> exporting data
string responses	24–28
echo	26–28
max count	25
termination response.....	25
Structure view	10–12
copy objects.....	12
deleting objects.....	11
moving objects	12
open objects.....	12
renaming objects	11
sub-objects.....	71–75
Summation object.....	93
Sync	29
synchronization with screen refresh	29
T	
Target file	119, 129, 130–33, 139
text file	<i>See</i> TXT file
TextDisplay object	54–59. <i>See also</i> E-Objects
font	57–58
properties	58–59
Toggle Bookmark.....	16
Toolbox	8, 18
Tree View.....	11
TXT file	213–14, 215–18
U	
Undo Merge	136
Unreferenced E-Objects folder.....	10, 12, 14, 18
User-Specified Parameters	<i>See</i> Startup Info
user-written script.....	3, 18, 83, 93
W	
Wait object	88–89. <i>See also</i> E-Objects
properties	88
WAV files	18, 84, 111, 207