User's Guide and Reference
MediaLab v2014

© 1995-2014 Empirisoft Corporation. All rights reserved.

MediaLab v2014 Users's Guide

© 1995-2014 Empirisoft Corporation

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of Empirisoft Corporation.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

Last Modified: March 2014, New York, NY.

Empirisoft Corporation
347 West 39th Street, 13th Floor
New York, NY 10018

Website: www.empirisoft.com

Technical support is available from support.empirisoft.com

Information regarding contact via regular mail, email and phone is available at www.empirisoft.com/company.aspx.

DirectRT, DirectIN and MediaLab are trademarks of Empirisoft Coporation.

Table of Contents

Part I MediaLab Overview **1**

Features ... 1
Setup ... 4
Key Concepts ... 6

Part II Main Program Window **8**

Run ... 9
Navigation ... 10

Part III Experiment Editor **11**

File ... 12
Edit ... 14
Options ... 15
Window ... 15
Printing ... 16
Editing Tips and Tricks ... 17

Part IV Experiment (.exp) Files **20**

Key Concepts ... 20
Overview Window ... 21
Condition ... 25
Position ... 25
File ... 26
BackGround ... 27
BackSound ... 28
BackVideo ... 28
Randomize Within Groups ... 29
Randomize Between Groups ... 30
Parameters ... 31

Part V Questionnaire (.que) Files **34**

Key Concepts ... 34
Overview Window ... 35
BackGround ... 39
BackSound ... 40
BackVideo ... 41
Filename ... 41
GoBack ... 42
Item Type ... 43
Item Wording/File ... 53

© 1995-2014 Empirisoft Corporation

Labels ... 55
Name ... 56
Omit Conditions ... 57
Parameters ... 57
Position ... 64
Randomize Within Groups ... 64
Randomize Between Groups ... 65
Skip To ... 66
Text Labels ... 67
Write to Data File ... 68
Write Data as Text ... 69
Write Reaction Time ... 69

Part VI Data **70**

Data FAQ ... 72

Part VII Samples **76**

Samples ... 77

Part VIII Preferences and Options **82**

Preferences and Options ... 83

Part IX Custom Items **88**

Repeating Custom Items ... 91
Single Vs. Multiple Variables ... 92
Variable Reference ... 94
Samples ... 97

Part X Advanced Features **99**

Calculated Scores ... 101
Complex Skips ... 102
Post-Session Reports ... 102
Tips ... 103

Part XI Tips and Notes **104**

Specifying File Paths ... 104
Sizing Images and Movies ... 105
Miscellaneous Features ... 106
Trouble Shooting Tips ... 109

Index **111**

© 1995-2014 Empirisoft Corporation

Getting Started

MediaLab Features 1 and Setup 4

Introduction and Key Concepts 6

Main Menu Commands 8

Designing Experiments and Questionnaires

MediaLab Experiment Editor 20

MediaLab Experiment (.exp) Files 34

MediaLab Questionnaire (.que) Files 34

MediaLab Preferences and Other Options 83

MediaLab Samples 76

Welcome to the PDF documentation for MediaLab v2014. You can search this documentation using Adobe Acrobat's search tool or by using the table of contents or index at the end of the document. We have designed this document for easy printing from Acrobat should you prefer a hard copy manual. Note that all of the information contained in this document can also be accessed via the interactive help system located in MediaLab's *Help* menu, or by clicking on the file named "interactive users guide" in the *C:\MediaLab\Help* folder after installation.

If you are new to MediaLab, we suggest you start by running the on-line tutorial from the *Help* menu of the main program window. You can also access a number of additional (zipped) samples 76 in the *C:\MediaLab\Samples* folder.

If you have a question that these help files do not answer, please visit our support site at support.empirisoft.com or visit www.empirisoft.com/company.aspx to contact us directly.

1.1 Features

Examples of General Features

- Define as many experimental conditions 6 as you like.
- Flexible randomization features.
- Randomize items within a single group 64 and the order in which groups of items 65 are presented.
- Randomization routines at both the experiment level 29 and within questionnaires 64.
- Simple and complex skip patterns 66. Base skips on any prior response, or even combinations of prior responses 102.
- Calculate scores and other variables 101 while a session is running--present them on screen as stimuli, use them in skip patterns 102, or in post-session reports 102.
- Create and optionally print summary reports 102 from each session with graphs, summary data, scale scores.

- Embed your own or third-party programs 44 within an experiment or questionnaire.
- Create your own custom questionnaire items 88 using standard HTML forms--MediaLab will intercept the posted data!
- Run DirectRT 43 or Inquisit 47 sessions from within your MediaLab experiment.
- Create multiple custom preference settings 84 with choices of fonts and display colors.
- Context sensitive help such that pressing F1 at any time will help you with what you are currently doing.
- Option of on-the-fly editing 83 of Word and WordPerfect documents—edit them as you run through your experiment.
- Resolution Independent 105. Compatible with screen displays from 640x480 to 1600x1200 and beyond.
- Easy-to-use intuitive interface--no programming code necessary for all but custom items 88.
- Compatible with most non-English fonts 85 (e.g., Japanese, Chinese, Arabic, Greek).
- Customize all standard feedback messages 86 and/or translate into alternate languages.
- Optionally run experiments from a CD-ROM or Server 9 on LAN.
- Communicate with external equipment 62 for trial synchronization (e.g., psychophys equipment).
- Receive input from external devices 107 such reponse boxes and precision keyboards.
- Run sessions automatically 9 from other programs and batch files.

Examples of Flexible Questionnaire Design

- Scale responses 50 with up to 12 closed ended responses.
- Multiple response 47 (select all that apply) responses.
- Fill-in-the blank 44 responses (free or restricted).
- Ranking 49 allows subjects to drag and drop words, phrases or pictures to sort them by any criterion.
- Essay-type 44 responses.
- Thought listings 51 and recall listings 49.
- Have subjects rate their own thought-listing 51 responses on any dimension.
- Easy text-based instructions 47 for any question.
- Voice Responses 51 with optional voice response times.
- Present HTML 45 and other web pages with active hyperlinks. Internet Explorer browsing engine can display anything that IE can handle.
- Track browsing behavior 52 as subjects navigate through local HTML files or the World Wide Web. Track every URL the subject follows and record in and out times.
- Present Microsoft Word 52 and Microsoft PowerPoint 48 shows within your experiment or questionnaires.

- Gather on-line continuous rating data 48 during audio and video clips.
- Skip pattern capability 66 when a particular closed ended response is given.
- Questionnaire items can be omitted 57 in specified conditions.
- Easily modify and re-use 34 your questionnaires in other experiments.
- Option of self-paced or time-limited responses 60.
- Allow subjects to return to previous questions 42 if necessary.
- Embed subjects' responses 53 in subsequent question wordings.
- Create multiple style configurations 84 (e.g., colors and fonts) for use within the same experiment or questionnaire.
- Can't find what you need? Design your own custom items 88 using HTML forms.

Example of Multi-Media Capability

- Present graphics 46 (.bmp, .jpg, and .gif).
- Present sound 50 files (.wav, .mp3).
- Present video 47 (.avi, .mpg, .mov, .rm, .ra, .asf and more).
- Present combinations of graphics 39, sound 40, video 41 and questionnaire items.
- Present HTML, Flash, java, ASP and any other browser compatible files 45.
- Optional parameters 57 allow you to set the screen location of images and movies as well as set the size of video clips all the way up to full screen.

Examples of Data Features

- Writes data 70 to an easy-to-analyze .CSV text file and SPSS .SAV file.
- Your choice of which variables 68 and to write to the data file.
- Smart data structuring so results from multiple conditions can be viewed in a single Excel or SPSS file. You can start analyzing your results immediately.
- Two sets of data files are produced, one organized by questionnaires 71, the other by variable names 71.
- Includes a utility to merge 73 .CSV and .TXT data files collected on different computers.
- Optionally write reaction time 69 data for any response to the data file.
- Optionally write data from multiple computers to a single folder on your network 83.

1.2 Setup

System Requirements

For most uses, MediaLab v2014 will run well on any PC system or Mac (e.g., via BootCamp), running Windows XP or Windows 7. At least 256 Mb of RAM is recommended. If large experiments and/or large multi-media files are going to be used frequently, then additional RAM is advisable though not necessary. Video cards with at least 16 Mb of video memory also are recommended.

It is recommended that you also install Microsoft Office on machines that will be running MediaLab. These programs allow MediaLab to present documents fully formatted by Word and slide shows prepared in PowerPoint and allow you to execute a number of features using the conditional logic of Excel. Although MediaLab will work fine without them, these programs perform cooperatively with MediaLab to produce some very impressive functionality. Microsoft's Internet Explorer version 7 or later is required in order for MediaLab to present custom items and other HTML formatted pages (whether stored locally or on the internet). Finally, it is necessary to have a spreadsheet application (e.g., Excel, SPSS) installed in order to view the data files produced by MediaLab (although the .txt and .csv data files can also be viewed in any text editor).

Backward Compatibility

If you are accustomed to using a previous version of MediaLab (v1998/v3-v2012), you should be able to continue using v2014 without having to learn anything new. v2014 is backward compatible so that MediaLab experiments from all previous versions *should* run the same as they always have without requiring modification. Of course, be sure to confirm compatibility for yourself with any old experiments prior to scheduling participants. As we continue to updated v2014 with new fixes and features, we will post the changes on our support site at:

http://www.empirisoft.com/support/forumdisplay.php/50-MediaLab-v2014-Version-History

Installation

To install MediaLab, download and run *ml2014.x.x.exe* (where *x.x* is the specific release) available from http://empirisoft.com/Download.aspx?index=4. If installing from a CD-ROM then simply double-click on the *ml2014.x.x exe* file located on the CD. The installation program will guide you through the set-up procedure. After the setup is complete, follow the security instructions below to gain authorization for the workstation to run MediaLab.

1.Note to Network Administrators:

Users will require *read/write access* to the MediaLab folder following installation. If they do not have read/write permission for the folder, the license check may fail and will cause a new code to be generated. The installation should grant these rights to all users automatically, but if not then they may need to be set as follows by someone with administrator-level access:

Step 1. When logged on as an *administrator*, this can be done by right clicking on the *MediaLab* folder and selecting *Properties* and then *Security* and ensuring that regular users have *Read/Write* permission. If you do not see a *Security* tab, then try *Step 2*.

Step 2. Open *Folder Options* in the Windows *Control Panel*. On the *View* tab, make sure *Use simple file sharing* is not checked. It should be one of the last options. With that unchecked, you should now get a security tab when checking the properties of folders. Try *Step 1* again—that should do it.

Security

To help control unauthorized distribution and use of MediaLab, it has been secured with ATX™ software protection. MediaLab will not run on any machine until this security has been disabled. To disable the security follow these steps:

- Install and run MediaLab.
- You will receive a message that the current workstation requires authorization to use the software, and a code number will be displayed.
- E-mail the code to sales@empirisoft.com.
- If you are an authorized user, you will receive a translation of this code which will be a registration number unique for the machine on which you have installed MediaLab (usually the same day). Then, re-start MediaLab and enter the six-digit code provided to you. MediaLab should now be fully functional. Important: ATX codes are unique to every machine. Be sure to get authorization for each machine that will be using the software. If you have any difficulty at all with this procedure, please contact us for assistance via sales@empirisoft.com.
- Trial users may request a code that will enable MediaLab to run for 21 days.

Alternative Solution for Verifying Licenses

Tired of entering all those authorization codes? If you belong to a school that has purchased a departmental site license then we can hard code your department's IP prefix into MediaLab so that ANY machine starting with that prefix (e.g., 123.456.789.*) will start up right away if the code 99999 is entered. Multiple patterns can be accommodated. Ask sales@empirisoft.com for details.

Reinstalling MediaLab

If you are upgrading to a new version you will be asked to uninstall the existing version. Experiments, data folders, preferences, and licenses will *not* be affected by uninstalling. Of course, we recommend backing up experiment data and any existing MediaLab files (e.g., any .que and .exp files) before making any changes to your system.

System Settings

The following are suggestions for your general system set-up:

True colors

To allow for the full range of colors usable by MediaLab, it is recommended that you use 16, 24, or 32 bit color. Some older systems are set by default to 8 bit, which provides for only 256 colors (colors or color ranges may appear as dots, patterns, or solid regions). To change these settings, open the *Display* properties in your computer's *Control Panel*, and select *Settings*. Select the highest color range possible. 24 and 32 bit color ranges are both capable of displaying close to the full range of colors detectable by the human eye.

Show file extensions

MediaLab frequently makes use of *file extensions*. These are the three-letters following the period on file names (e.g., *.exe*, *.doc*, *.mpg*). For this reason, it is advisable that you don't instruct Windows to hide these from you. To set this as a default:

- Open any folder and select *Tools* from the pull down menu. Select *Folder Options*, and

then *View*. Deselect the option to have Windows *Hide File Extensions*. It is also useful to have Windows *Display the Full Path in Title Bar* as well as *Smooth Edges of Screen Fonts*. The latter setting should be on by default in Windows 7. In *XP*, the option can be found in *Control Panel > Display > Effects.*

Task bar NOT always on top.

It is possible to have the windows task bar always on top. This is not recommended for use on machines that will be running MediaLab as it gives participants easy access to the system, and it can obstruct some important features of MediaLab. To deselect this option, right click on the task bar and then click on Properties. *Always on top* can be de-selected here if you wish. Note the key sequence *Ctrl+Esc* will make the task bar re-appear at any time.

Screen Resolution

Although MediaLab was designed to work in all screen resolutions, how things look may depend on the resolution you choose. Certain items may need to be repositioned with the left and top parameters when running in high resolution modes. If you are working on different computers, it is often best to pick a single resolution for your experiment (e.g., 1024x768). To change these settings, open the Display properties in your computer's Control Panel, and select *Settings* (Windows XP) or right click on your desktop and select *Screen Resolution* (Windows 7).

1.3 Key Concepts

The Basic Idea

You need three things to conduct a MediaLab experiment. First, you need the materials or stimuli you want to show participants. In many cases, you may simply have a number of instructions and questions you want to give participants. Although MediaLab has the capability to present many file types, some people just want to administer a basic questionnaire. If this is the case, then you should find MediaLab an extremely easy way to do this. However, if you want to show images, play sound files or movie clips, display Word documents, PowerPoint shows, HTML pages, or execute other stand-alone programs, you will find MediaLab to be extremely cooperative in this respect as well.

Second, you need dependent measures. Typically, these will consist of questions you have regarding participants' responses to your instructions or materials. These questions may consist of lengthy open-ended responses, short fill-in-the-blank type responses, multiple choice and scale responses, thought or recall listings, or even ratings of subjects' own responses. MediaLab will allow you to ask your questions using a wide variety of question formats.

Finally, you need an experimental design. For this, you need to know which of your materials and questions are assigned to each of your experimental conditions, and the order in which they will presented. You may have just a single condition with a single questionnaire, or a complicated factorial design with many questionnaires and randomization. Either way, MediaLab is very flexible in terms of experimental design.

Once you have your design planned, your questions in mind and your materials prepared, MediaLab plays the simple role of executing your experiment and preparing your data for analysis. To get MediaLab to do what you want, you need to understand two new types of files:

The Experiment File

The experiment file (i.e., files with the *.exp* extension) deals with things on a very general

level. Here, you simply specify which files are going to be presented in each of your conditions. You basically tell MediaLab, "In condition 1, present file1, file2, file3, etc., and in condition 2, present file1, file4, file5, etc." These files can be Word documents, PowerPoint shows, HTML files, movies, sounds, images, executables, and MediaLab Questionnaires. To learn more about experiment files, see the page about Key Concepts of MediaLab Experiment Files 20.

The MediaLab Questionnaire

This is where MediaLab becomes extremely flexible. The questionnaire (*.que*) is a file that can be presented just like any other in your experiment. In the experiment file, you identify the files to be presented in your various conditions—including questionnaire files. Questionnaires differ from the other file types in that they are a single self-contained file full of instructions for MediaLab. These instructions can tell MediaLab to present various files (e.g., images, sounds, videos, executables, Word documents, PowerPoint shows, HTML pages), but questionnaires also contain the instructions for MediaLab to ask questions and gather data.

MediaLab questionnaires provide instructions for essay-type open ended responses, multiple choice responses (with single or multiple response options), multiple response questions (e.g., select all that apply), fill-in-the-blank answers, continuous on-line ratings, thought and recall listings, and thought ratings (where subjects rate there own open ended responses on dimensions you define). To learn more about questionnaire files, see the page about Key Concepts of MediaLab Questionnaire Files 34.

Running the Experiment

1. Once you have defined your experimental conditions, and created the files which it is going to present, simply click the *Select and Run Experiment* command from the *Run* menu of MediaLab. After finding and double clicking on your experiment, you will be asked which condition you want to run, and what subject ID you want written to the data file. The rest is easy. MediaLab runs the experiment, gathers the data and writes it to all to a single data file that can be read directly by Excel) or SPSS (unlimited variables). MediaLab also creates a data input list for your data set in both Excel and SPSS formats so that you can start analyzing your data immediately. MediaLab also provides a data merging utility 73 so that you can easily combine data gathered on different computers before running your analysis. See Data 70 for more details.

What MediaLab is *NOT*

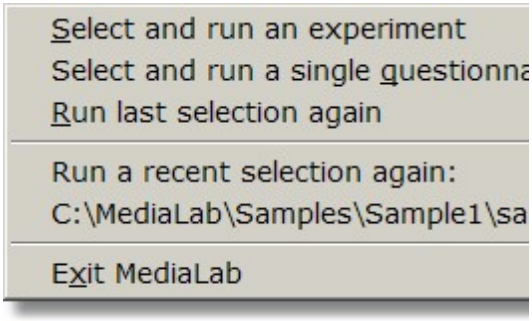
Before planning a study with MediaLab it critical to know its limitations. Most importantly, we advise serious caution about using any program written for the Windows environment for studies requiring extremely accurate presentation times and response times involving visual images. MediaLab does utilize a timer with a resolution of less than one millisecond and was originally written with the functionality to conduct a wide variety of standard reaction time experiments such as priming studies. For a number of technical reasons, this capability has been removed from the main program and has been developed as a stand alone program called DirectRT that will work cooperatively with MediaLab (although we still use the high-res timer). DirectRT provides high precision timing routines for the presentation of images and measurement of response times in the Windows environment and is recommended for any application where high precision timing is critical.

The second limitation of MediaLab to keep in mind is that it works on the principle of a *static* experimental design. Dynamic changes in design are not currently possible. Basically this means that a subject needs to be assigned to a condition at the start of the experiment, and what occurs in that condition is fixed. Although fairly complex skip patterns and dynamic stimuli can be set up in MediaLab questionnaires, a subject can not be re-assigned to a different condition based on a response contingency. This is not usually a big deal, but it's something to keep in mind.

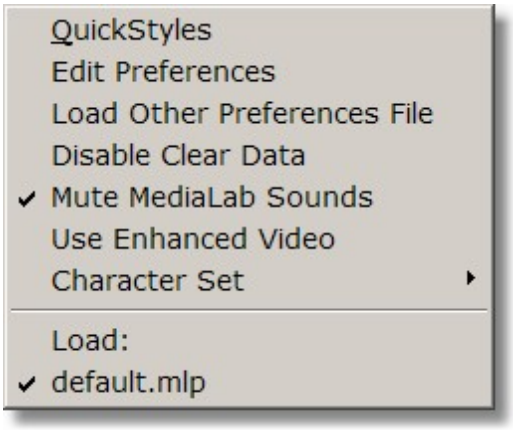
Menu Commands

The main menu commands are found at the top of the MediaLab program window. These are the general commands used for running experiments, viewing data files, and for setting preferences. Note that the MediaLab Experiment Editor 11 has it's own separate menu system.

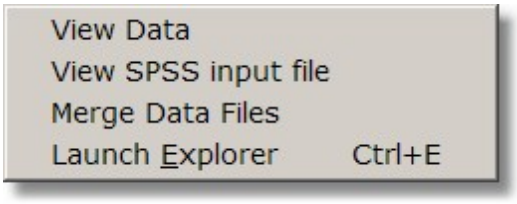
Run 9



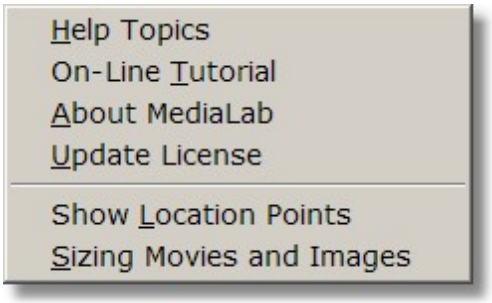
Preferences 82



Data 70



Help 1

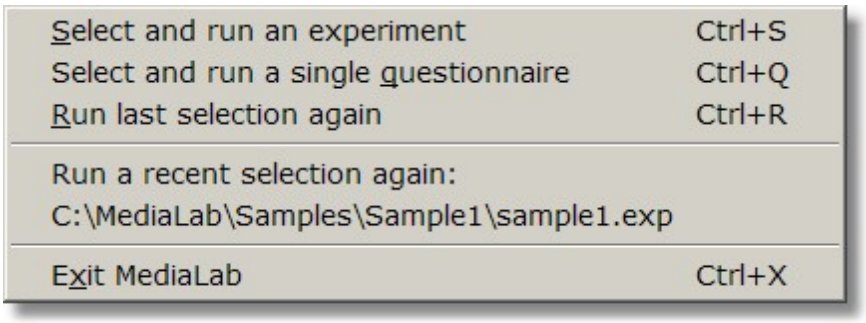


Experiment Editor 11

2.1 Run

Menu Commands

When you are ready to run your experiment, either as a trial run or to collect real data, you have three options. From the *Run* menu you can choose to:



Select and run experiment

If you haven't run the experiment before, this choice allows you to search through the folders on your machine to find the desired experiment. Locate the experiment file and then either double click on it, or select it and click *Open*.

Select and run a single questionnaire

If you want to test or run a single a questionnaire or just see what it looks like you can click on *Select and Run a Single Questionnaire*. MediaLab will automatically create a single item experiment file 6 and run it. All data files will be located in the usual place but will be named after the questionnaire, e.g., *self-esteem_que.csv*

Run the most recently executed session

This choice will execute the experiment or questionnaire most recently run on the machine. The key sequence *Ctrl-R will* perform the same function. This command is handy when multiple participants are being run back-to-back in the same session.

Select from the 5 most recently run sessions

MediaLab keeps track of the five most recently run sessions and displays them in the *Run* menu. Clicking on any one of them will run the selected session (assuming that the displayed pathway is accurate).

Running an Experiment from a CD-ROM

By default, MediaLab writes data to the same folder that contains the experiment files. This prevents you from running experiments off of a CD-ROM since you can't write data to a read-only CD-ROM. To accomplish this, you can specify an alternate data folder in your preferences file 83. So long as the alternate data folder is one to which MediaLab can write the data, then running the experiment off a CD is no problem.

Running Experiments Over a Network

In the same way that you can write data to a folder on another computer, you can also run an experiment from another computer. To do this, again, you must *map the drive* of

the other computer which contains the experiment (see above). Once you have done this, you can run the experiment as if it were located locally. From the *Run* menu, choose *Select and Run an Experiment* or *Select and Run a Single Questionnaire* and browse to the experiment located on the mapped drive (e.g., *h:\experiment1\myexperiment.exp*). This way, multiple computers can access the same experiment from a single computer. As with writing data to an alternate data folder 83 (see above), it is recommended that you specify a unique machine code in Preferences 84 because all the data from all machines will be writing to the data folder on this single computer. This option may not be suitable for experiments that require a lot of bandwidth (e.g., those that use a lot of video and sound) or for experiments using many computers on a slow network.

Running Experiments From Another Program, Shortcut or Batch File

In some cases you may want to start an experimental session from another program or shortcut without any of the usual MediaLab startup screens. You can begin a MediaLab session from within a Visual Basic application or program with VBA support by using a shell command as follows:

```
Shell "c:\medialab\medialab.exe c:\medialab\samples\sample1\sample1.exp,15,1"
```

If you specify the subject and condition IDs like this (i.e., "15,1" in this example) then the session will start automatically with no prompts and will end upon completion. Alternatively, you may specify "0" for either or both. If you specify a "0" for both (i.e., "0,0") then MediaLab will attempt to use your autostart file if it finds one. If one does *not* exist, then MediaLab will prompt you for the IDs. You may also specify one but not the other, in which case MediaLab will prompt you for the unspecified value.

This can also be run from the Windows *Run* command, a *batch* (.bat) file or *shortcut* file--notice the difference in quotation marks:

```
"c:\medialab\medialab.exe" c:\medialab\samples\sample1\sample1.exp,15,1
```

2.2 Navigation

Key Navigation

As an alternative to having subjects click on the *Continue* and *Go Back* buttons, you can have subjects use the keyboard to navigate their way through your experiment. The *Backspace* and *Space bar* keys will simulate mouse clicks on the *Go Back* and *Continue* buttons, respectively. This does not apply to *essays*, *fill-in-the-blanks*, and *thought-listings* which can still be ended with the *Escape* key as an alternative to the mouse click (since the *Space bar* and *Backspace* keys are necessary for open ended responses). Note: If you use an *Executable* item, the navigation keys will become active only after the executable program has ended and control is returned to MediaLab.

Not: When using an *HTML* item in your questionnaire that allows for spacebar or backspace input, you may want to add a k-1 parameter 60 to disable keyboard navigation for that item. This does not apply to *Custom* items made with HTML because custom items do not allow for keyboard navigation.

Secret Keys

When test-running an experiment, you may want to skip over various items, files, or questionnaires that you already know are working. You also may have timed items that you don't want to wait for while you're testing things out. For this reason, there a few secret key combinations that will allow you skip around. All use the right or left arrow keys on your keyboard in combination with either or both of the *Alt* key or *Ctrl* key. Press

each key down and hold it until each of the remaining key(s) is pressed.

Ctrl + Left Arrow	*Go back to previous question in a questionnaire*
Ctrl + Right Arrow	*Skips to next question in a questionnaire*
Shift* + Alt + Right Arrow	*Quits the questionnaire or other current file*
Ctrl + Alt + Right Arrow	*Quits the experiment in progress*

Notes

*In previous versions, the combination *Alt + Right Arrow* would end the current questionnaire or file. Because this combination is now commonly used in web browsing we added a *shift* key to the combination to avoid accidental endings.

Some item types that involve videos and or HTML may require that you first click on the outer border of the MediaLab space in order to use the secret keys. This is because the Windows Media Player and Internet Explorer engines will sometimes intercept keystrokes and won't share them with MediaLab. For the same reason, the space bar may cause a video to pause rather than causing MediaLab to continue to the next item. In any case, a click on the outer border will return keystroke input to MediaLab if and when this happens.

Due to setting of some Intel and NVidia brand chip sets, use of *Ctrl + Alt + the arrow keys* may rotate your screen orientation instead of quitting the experiment in progress. To disable the screen orientation in Windows 7, right click anywhere on your desktop and select Graphics Options > Hot Keys > Disable.

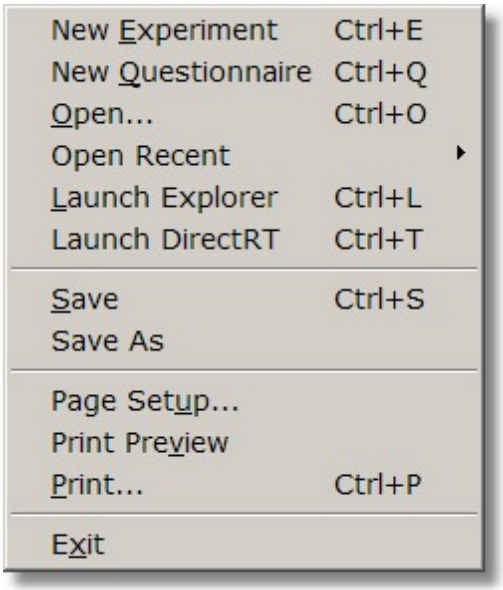
StatusBar

The StatusBar is displayed at the bottom of the main MediaLab window. Allows you to view some convenient information about the experimental session. The first cell in the status bar displays information about current activities and when nothing else is going on, it displays the name of the last experiment run during the current session. The next cell displays the last *Subject ID* and *Condition* run during the current session. The next two cells display the current date and time. Can be hidden with option on the Tools menu.

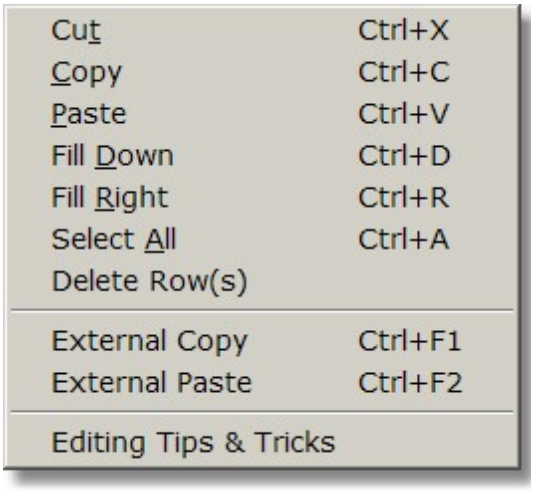
Menu Commands

These main menu commands are found at the top of the *Experiment Editor* window. To launch the experiment editor, either click on *Experiment Editor* in the main MediaLab window, or click the *MXEdit* icon in the MediaLab startup folder. These are the general commands used for editing Experiment (*.exp*) and Questionnaire (*.que*) files.

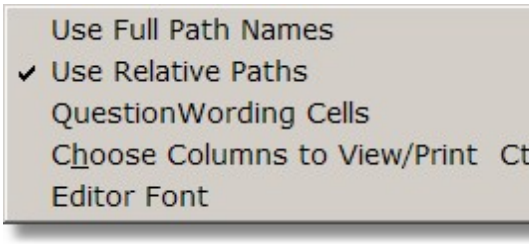
File 12



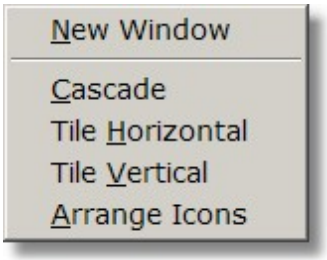
Edit 14



Options 15



Window 15



3.1 File

QuickInfo

The following describes the functions provided in the File menu of the MediaLab Experiment Editor.

Commands

New Experiment

Creates a new experiment file for editing. See Experiment Files 20 for details.

New Questionnaire

Creates a new questionnaire file for editing. See Questionnaire Files 34 for details.

Open

Opens an existing experiment (*.exp*) or questionnaire (*.que*) file for editing.

Open Recent

Select from the list of recently edited experiment and questionnaire files for editing.

Launch Explorer

Runs Windows Explorer for file browsing. See suggestion regarding the visibility of file extensions 5 (e.g., .que, .exp, .htm, .doc, .etc)

Launch DirectRT

Runs *DirectRT* if you have it installed. See www.empirisoft.com/directrt.aspx for details.

Save

Saves the active file with its current name and location on your drive.

SaveAs

Saves the active file but allows you to specify a new file name and/or save location.

Page Setup

Allows you to set print options such as margins, portrait vs. landscape and which columns to print. See Printing 16 for details.

Print Preview

Allows to preview printed output.

Print

Prints the contents of the active file.

Exit

Exits the MediaLab Experiment Editor.

3.2 Edit

QuickInfo

The following describes the functions provided in the Edit menu of the MediaLab Experiment Editor. See also Editing Tips and Tricks for a variety of editing shortcuts.

Commands

Cut

Deletes the selected text or cells and places the deleted contents in the MediaLab clipboard for pasting elsewhere.

Copy

Copies the selected text or cells and places the copied contents in the MediaLab clipboard for pasting within the same or other questionnaire. See also *External Copy* below.

Paste

Pastes the contents of the MediaLab clipboard to the selected cell(s). See also *External Paste* below.

Fill Down

When cells from multiple rows are selected, copies the value from the top selected cell(s) to the selected cell(s) below.

Fill Right

When cells from multiple columns are selected, copies the value from the left most cell(s) to the selected cell(s) to the right.

Select All

Selects all the cells in the active file.

Delete Rows

Deletes the selected rows from the active file.

External Copy

The MediaLab experiment editor uses it's own private clipboard for cutting, copying and pasting. If you want to copy cells to paste in an external application (e.g., Excel, Word, a text editor), select the cells and choose External Copy.

External Paste

To paste cells or text from external applications (e.g., Excel, Word, a text editor), select External Paste.

Edit Tips and Tricks

Hints and short cuts 17 for working in the Experiment Editor.

3.3 Options

QuickInfo

The following describes the functions provided in the Options menu of the MediaLab Experiment Editor.

Commands

Use Full Path Names

When selecting files by double-clicking in a cell, this option specifies that the full path be included (e.g., *c:\experiments\exp1\images\image1.bmp*). See Specifying File Paths 104 for details.

Use Relative Paths

When selecting files by double-clicking in a cell, this option specifies that the path be specified relative to the experiment (*.exp*) file (e.g., *images\image1.bmp*). See Specifying File Paths 104 for details.

Question Wording Cells

Expand All – Shows the complete contents of all question wording cells in the overview window.

Collapse All – Shows only the first line of each question wording cell. Clicking on any question wording cell will automatically cause it to expand.

Choose Columns to View/Print

Allows you to hide or show columns in the overview window. This feature is especially handy if you wanted to print only a select number of columns from the overview window.

Editor Font

Allows you to choose an alternate font for editing text within the editor. Can be useful for international font requirements. You may also need to choose a *Character Set* for your language from the main MediaLab *Preferences* 85 menu

3.4 Window

QuickInfo

The following describes the functions provided in the Windows menu of the MediaLab Experiment Editor.

Commands

New Window

Equivalent to selecting *Open* from the *File* menu.

Cascade

Neatly arranges all open windows in an overlap pattern making all window titles visible.

Tile Horizontal

Stretches all open windows their maximum width and arranges them in a horizontal overlap pattern.

Tile Vertical

Stretches all open windows their maximum height and arranges them in a vertical overlap pattern.

Arrange Icons

Neatly arranges all minimized window icons.

Hint

The most useful function in the Windows menu is that at the bottom of the menu will be a list of all currently open files. You can select which file to bring to the forefront by clicking on any file in this list. Note also that you can use *Ctrl+Tab* to flip through all open files.

3.5 Printing

QuickInfo

Prints the content of your experiment or questionnaire file.

Value

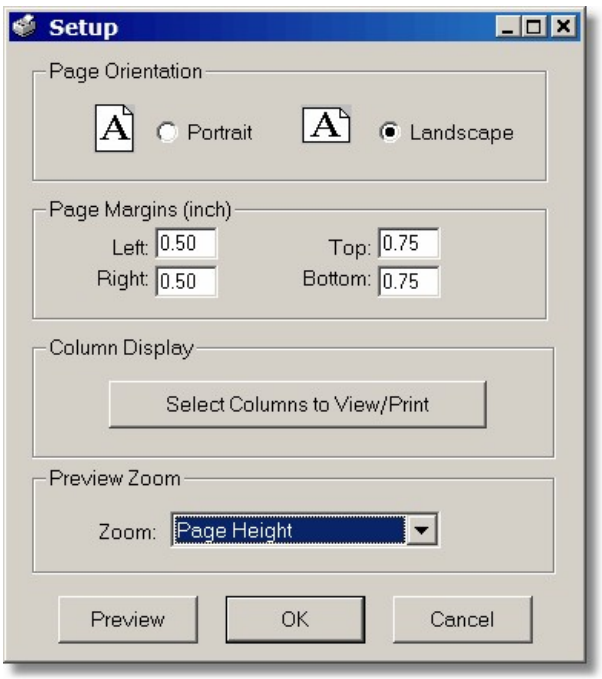
Mouse-click, Alt-P (Overview editor only)

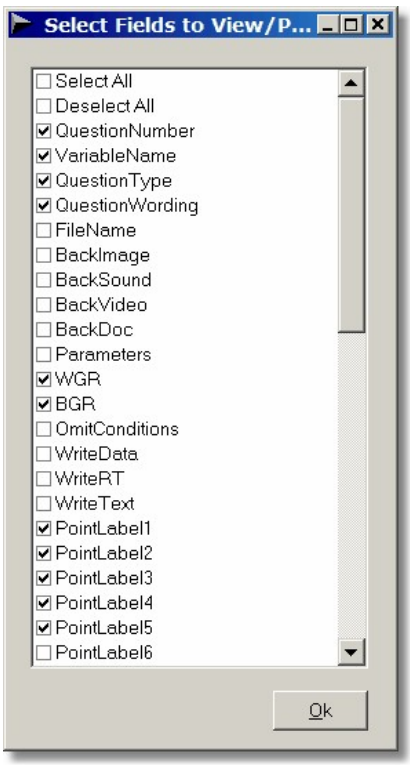
Purpose

You may wish to print a hardcopy of information in your experiment or questionnaire file. To select which columns to print, click *Select Columns to View/Print*. Check all the columns you want to print. Note that you can start by clicking *Select All* or *Deselect All*.

You can manually widen any column if you want more of it displayed when printing.

For question wording cells (questionnaire files only), you can expand (print all lines) or collapse (print only the first line) the cells when printing. When selecting *Print*, select *Expand All* or *Collapse All* for the *Multi-Line Cells* option.





3.6 Editing Tips and Tricks

QuickInfo

The following describes how to use some of the editing features of MediaLab's new Experiment Editor:

Moving Around and Editing a Cell 18

Copying a Single Cell 18

Drag & Drop (Turbo) Copy 18

Copy a Block of Cells 18

Copying an Entire Line/Record 18

Copying Multiple Lines/Records 18

Inserting Copied Lines/Records 18

Filling Down & Filling Right 19

Double-Clicking Functions 19

Relative vs. Absolute Paths 19

Expanding Question Wording Cells 19

Selecting Cells to View 19

Copying and Pasting to and from Other Applications 19

Editing Files Directly in Microsoft Access 20

Tips and Tricks

Moving Around and Editing a Cell

The arrow keys will allow you to navigate through the spreadsheet. To select a cell in order to edit the contents, press the *Enter* key. When you are finished editing the cell, press *Enter* again, and the arrow keys will again let you navigate. You can also leave a cell while editing it by using the *Up* or *Down* arrow keys. To add a hard return/line feed to your question wording, do this in *Details* 36 mode.

Copying a Single Cell

To copy a single cell, move to that cell select *Copy (Ctrl+C)*, then move to the desired target cell and select *Paste (Ctrl+V)*. Only entire cells may be copied & pasted in the main overview spreadsheet. To copy and paste only portions of cells, do this in *Details* 36 mode.

Drag & Drop (Turbo) Copy

You can quickly copy the contents of one cell to another using the right-mouse button. Simply right-click on the cell you want to copy and drag it to the desired cell. Only one cell at a time can be copied this way. This can also be done between multiple open files (i.e., drag from a cell in one file and drop to a cell in another file).

Copy a Block of Cells

To copy a block of cells, highlight the block and select *Copy (Ctrl+C)*, then move to the upper left cell where the pasting should begin and select *Paste (Ctrl+V)*. You can also copy a block of cells from one file and paste it into another using this method.

Copying an Entire Line/Record

To copy an entire line or record, use the record selector at the very left of the spreadsheet (this selects all the cells in that row). Select *Copy (Ctrl+C)*, then move to any cell on the line where you want it copied and select *Paste (Ctrl+V)*. This will copy the entire record, including its details, to the new line. To insert copied lines, see *Inserting Copied Lines/Records* below. You can also copy an entire record from one file and paste it into another using this method.

Copying Multiple Lines/Records

Follow the instructions for copying a single record (above), but highlight as many rows as you like with the record selector. Move to any cell in the row at which you want to start the pasting, and select *Paste (Ctrl+V)*. You can also copy entire records from one file and paste them into another using this method.

Inserting Copied Lines/Records

1. Note that pasting overwrites the cells in the current row. Therefore, to insert the copied line(s), without affecting the other items in the file, go down to the first blank line, and then select *Paste (Ctrl+V)*. To move it to the desired location, simply assign it a position number and then click Sort 37. You can also copy entire records from one file and insert them into another using this method.

Filling Down & Filling Right

Sometimes you may want to copy one cell over and over to a series of cells below it or to its right. An example would be that you want to copy the same parameter value or scale point labels to 20 sequential items without having to retype it each time. With the *FillDown* function, this is easy:

- To copy a cell to a series of cells below it, highlight the block of cells (including the one you want to copy) and then select *FillDown (Ctrl+D)*. This will copy the top cell to all the cells selected below.
- To copy a cell to a series of cells to its right, highlight the block of cells (including the one you want to copy) and then select *FillRight (Ctrl+R)*. This will copy the top cell to all the cells selected below.
- Note that the filling functions only operate in the main overview spreadsheet and not in *Details* mode.

Double-Clicking Functions

If you double click on the *Position*, *Name*, or *Condition* (*Experiment* files only) columns, this will immediately take you into *Details* mode. If the item type is a file, then double-clicking on the *FileName* or *QuestionWording* column will immediately open up a file browser window so you can select your file. If you select *Cancel*, this will return you to *Edit* mode so you can manually edit the cell. Double clicking on the *BackGround*, *BackSound*, and *BackVideo* columns will also immediately open a file browser window so you can choose an appropriate file type.

Relative vs. Absolute Paths

To use *Absolute* or *Relative* (default) paths, select the appropriate option from the *Options* Menu. What does this mean? When you select files using the file browser, you can opt to have it return the entire (aka *absolute*) path of the file (e.g., *c:\experiments\steve\exp1 \images\image1.bmp*), or the path relative to the experiment (e.g., images\image1.bmp). Relative paths are recommended because the experiment folder (e.g., in this case exp1) can be moved anywhere on any computer and in any folder and MediaLab will know where to find the file. The only time an absolute path is necessary is when the desired file is not located somewhere within the experiment folder. However, if you are using Relative Paths, the editor will enter an absolute path if this is the case.

Expanding Question Wording Cells

When a question wording cell uses multiple lines, the cell will expand to show this when you click on it. If you would like to see all such cells expanded at the same time, select *QuestionWording Cells > Expand All* from the *Options* menu. This is also an option you can select when printing your file.

Selecting Cells to View

When working with your file, you may wish to view only certain columns. To select the columns to view, select *Choose Columns to View/Print (Ctrl+H)* from the *Options* menu. This is also an option you can select when printing your file. Note that the column selection feature applies only while the file is open and is not saved for future use.

Copying and Pasting to and from Other Applications

The new editor uses its own proprietary clipboard so you can not use the usual copy and paste functions to copy cells to and from other applications (e.g., *Word*, *Excel*, etc.). To do this use the *External Copy* and *External Paste* commands on the *Edit* menu. These

functions access the usual Windows clipboard.

Editing Files in Microsoft Access

Questionnaire (*.que*) and Experiment (*.exp*) files are really just small Access database files. If you right click on any questionnaire or experiment file and select *Open With*, you can choose *Microsoft Access* if you have it installed (you may have to browse for it). If you do this you may be informed that you can not make changes to database objects. That's ok. You may also be asked if you want to convert the database to a more recent version--just say no, and ask to open the file. Once the file is open in Access, you can double click on *Table1* to edit the file. This allows you to bypass the MediaLab Experiment Editor completely or you can do some of the work in Access and some of the work in the MediaLab editor. Some people, for example, like to load their files into Access for heavy copying and pasting jobs, or to use the Access Spell Checker.

Key Concepts 20: Click here to find out more about exactly what the Experiment Files are and do in MediaLab

Overview Window 21: Click here to see an example of an Experiment File and find out more about its basic features.

File Details

Click on any of these links to find out more about specific features of the Experiment Files in MediaLab

Condition 25
Position 25
File 26
BackGround 27
BackSound 28
BackVideo 28
Randomize Within Groups 29
Randomize Between Groups 30
Parameters 31

4.1 Key Concepts

The experiment editor allows to you to create, view and edit the details of your experimental conditions. The experiment file you create using this editor is the first thing that MediaLab looks at when you go to run the experiment. You specify here which questionnaires, images, videos, sounds, documents, html files, executables or PowerPoint shows you want to present in each condition, and the order in which you want them presented. When you run the experiment, MediaLab will ask you which condition you want to run. It will read your experiment file, determine which files you have specified for that condition, and then it will present those files to the subject in the order you have specified.

Think of the Experiment Files like a tabletop and the Questionnaire Files as a bunch of jigsaw puzzle pieces. The Experiment Files serve as the surface upon which you arrange all of the individual pieces of your experiment--Questionnaire Files. The Questionnaire Files serve the all-important purpose of where you *organize* the general order of elements of your experiment like the items and their presentation order.

With the exception of questionnaires, *no data is gathered for files that you specify here*. This is a very important concept. Data is only gathered in questionnaire files that you create. Questionnaires are much more flexible in terms of the types of files and information that can be presented and the type of responses and data that can be collected.

The Experiment file that you create with the experiment editor simply tells MediaLab which files are to be presented in each of a virtually unlimited number of conditions. This can be as simple as assigning a single questionnaire file to each condition, or as complicated as multiple questionnaire files assigned to each condition, each interspersed with video, audio, graphics, Word documents and PowerPoint shows for which data does not need to be gathered (e.g., for instruction screens or debriefing documents).

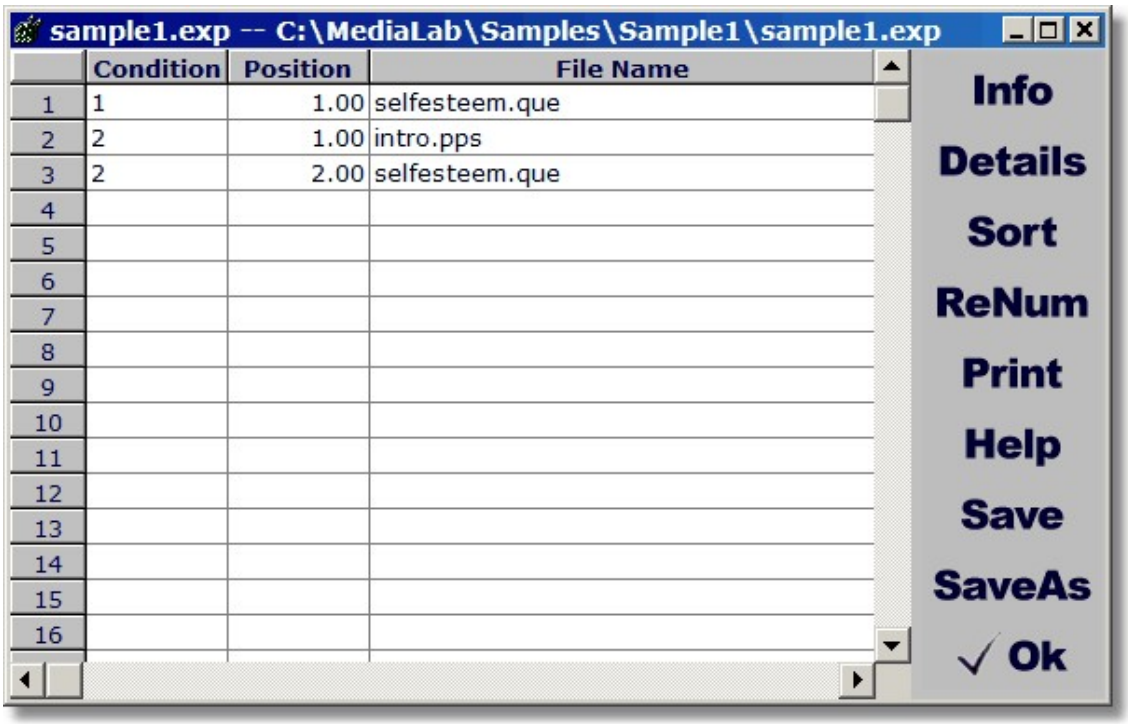
Remember though that all file types can also be presented in a questionnaire if you like. That is why the experiment file can be as simple as stating which questionnaire (or set of questionnaires) is assigned to each condition. Additional files can be specified simply for the purpose of having this flexibility in the case that your application requires it.

4.2 Overview Window

QuickInfo

In the overview editor, you can scroll through the basic information regarding your experiment. In this mode, you can see and define which files are presented to subjects in your different experimental conditions, and the order in which they're to occur.

Click on a specific area for help on that topic:



Hints

Double-click on any condition or position field, and the file details will be displayed for the file at that location. Double-click on any file field, and you will be able to search your drives for the file you're looking to put there. If you want to edit the file name field without being prompted for a file, just highlight the cell and press enter to edit it.

Info

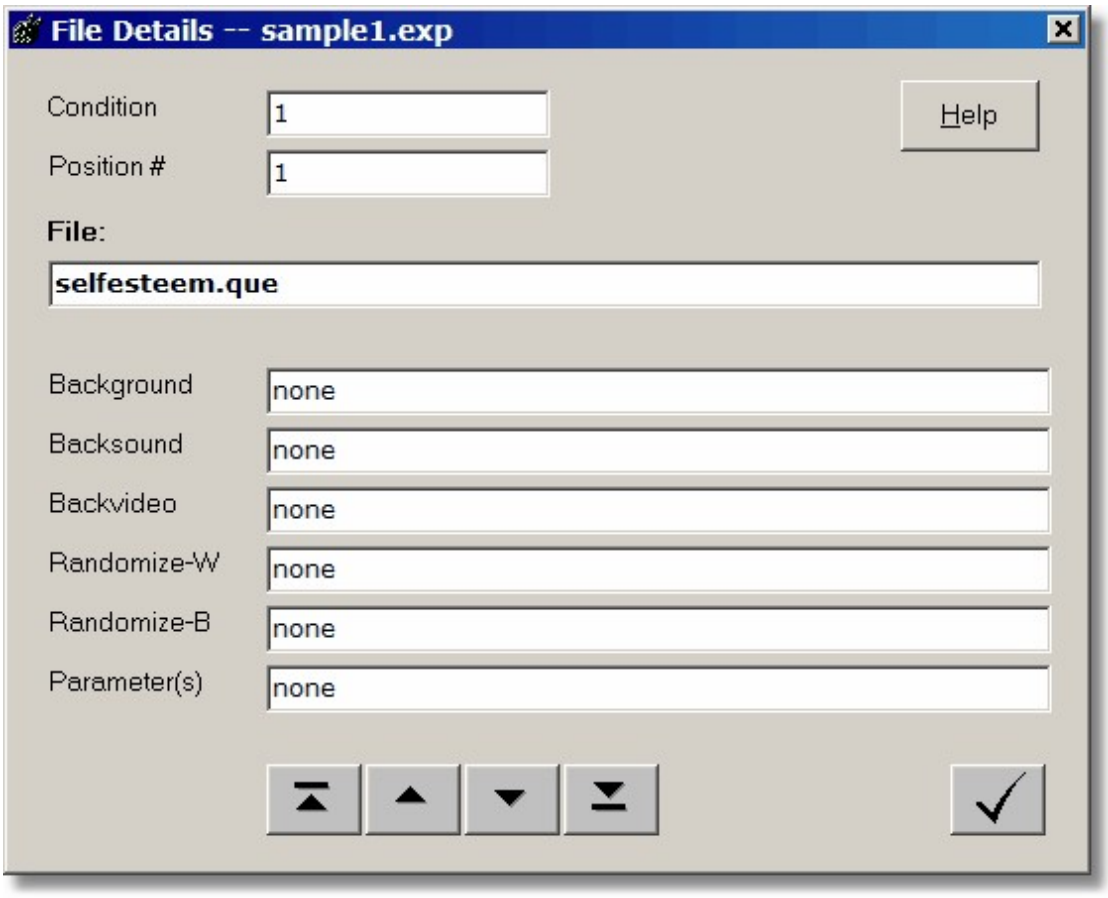
Allows you to save summary information and detailed notes regarding your experiment.

The title and comments are for the Editor's reference only. The comments field has a very large capacity if you wish to exploit it.

Details

Allows you to set various options for the presentation of your experimental files.

Explore a specific area for help on that topic:



Sort

Sorts the files in the experiment, first by condition and then by position.

The sort button makes life a little easier by automatically reordering all of the files in your experiment so that you need not worry about where you are placing a file at any given time. When MediaLab sorts the experiment files, it will first place all of the files in each condition together. Next, it will reorder the files within each condition by their ordinal position. If you forget to do this, the editor will do it automatically when you exit. You can manually sort whenever and as often as you like by clicking the sort button.

Hints

To add a new file between two existing files (e.g., at positions 2 and 3), assign it a position value between these values (e.g., 2.5) and then click the *Sort* button. After sorting, you can automatically renumber all of your items as sequential integers by clicking the renumber 23 button.

Important: MediaLab will first sort by the condition 25 field and then by the position field, so be sure that you identify the condition of the file before you sort.

Renumber

Renumbers the position 25 values within each condition 25 with sequential integer values.

The renumber button cleans up your position numbering by replacing the values with sequential integers. Within each condition, the first file will get a position value of *1*, the second will get a *2* and so forth. This is not a necessary operation and is provided mainly for the purpose of aesthetics and to help you verify that you have the correct number of files in each condition.

Hint

If your files are out of order, be sure to sort 23 before you renumber.

Print

See Experiment Editor, Printing 16

Help

Starts the interactive help system.

You can search the documentation using the Contents, Index and Search tabs in the left-hand window. Also note that MediaLab takes advantage of context sensitive help such that pressing F1 at any time will often direct you immediately to help with what you are working on. A printable PDF copy is available in the C:\MediaLab\Help folder.

Save

Saves the current file.

Save As

Saves the current file while allowing you to set the file name and/or location.

OK

Closes the *Details* window and returns to the *Overview* window.

Navigation

There are many ways to navigate your way around the experiment editor. The most useful methods to be aware of are:

- to jump from field to field, you can simply click on the field you want to edit, or
- you can also use the *tab* key and the arrow keys to jump from one field to the next.

When your are editing in the overview window, double-clicking will allow you to edit any field or will pull open a file search box if a file is required. If you want to edit a field that requires a filename but do not want to open a file search dialog, then simply highlight the field and press enter to edit the field directly.

All forms have a button with a check mark, like this: . This button closes whatever editor you are currently in. You will always be given the option to save any changes that you've made.

Details Editor

In the details editor, you will notice four buttons with arrows:

- The first key will take you to the details of the first file in your experiment,
- The second key takes you to the file immediately preceding the one you are currently editing,
- The third key takes you to the file immediately after the one you are currently editing,
- and the fourth key takes you to the last file in your experiment.

4.3 Condition

QuickInfo

Identifies which files are presented together in an experimental session.

Values

Any text. Usually numbers but words are allowed. For more than 9 conditions, you can label the first 9 conditions using two digits (e.g., *01*, *02*, *03* etc.) to preserve numeric/ alphabetic consistency.

Purpose

When you run a MediaLab session, you will be asked for two things: 1) a *Subject ID* for writing to the data file, and 2) the *Condition* you want to run. MediaLab will then read your experiment file and look for all the files that belong to the condition you specify. Therefore, in the experiment editor you need to identify which conditions receive which files.

In your experiment file, you can create as many conditions as you like. Simply create a unique identifier for each. You can use numbers (*1*, *2*, *3*, and so forth) or text (*cond1*, *male speaker*, *sg24*, and so forth) to identify your conditions. Just be sure to give the same condition id to every file that you want to be presented in that condition. If a file is to be presented in multiple or all conditions simply list it once with each condition id.

Hints

Values are treated as words for purposes of sorting. So if you have more than nine conditions and are using numbers to label the conditions you may want to use two digits (e.g., "01", "02" … "09", "10" etc.) to maintain your numerical order after sorting.

Using long text identifiers can be a pain when you are running many subjects and have to type the long condition names to start the experiment every time. Using numbers or short letter combinations to identify your conditions will solve this.

For factorial designs, try assigning single letters to represent the various levels of your factors. For example, imagine a persuasion experiment with a 2 (attractive vs. unattractive speaker) x 3 (strong vs. weak vs. no arguments) design. The two levels of the speaker factor could be represented with A and U, and the three levels of the argument factor could be represented with S, W, and N, respectively. The six conditions might be identified as *AS*, *AW*, *AN*, *US*, *UW*, and *UN*. Or for simplicity, you might simply label them *1*, *2*, *3*, *4*, *5*, and *6* while keeping a record of which number corresponded to which experimental condition.

4.4 Position

QuickInfo

Identifies the order in which files are presented in each condition.

Values

Any number. Decimal values are allowed.

Purpose

This field determines the order in which files are presented in each condition. MediaLab will automatically sort the files in each condition by the values you specify in the *Position* field. You can manually sort the file at any time by using the Sort 23 button.

Hints

To add a new file between two existing files (e.g., at positions 2 and 3), assign it a position value between these values (e.g., 2.5) and then click the *Sort* button. After sorting, you can automatically renumber all of your items as sequential integers by clicking the Renumber 23 button.

Important: MediaLab will first sort by the condition 25 field and then by the position field, so be sure that you identify the condition of the file before you sort.

4.5 File

QuickInfo

Names of the files to be presented in each condition

Value

Any file name ending with the following three letter extensions:

.que	MediaLab Questionnaire 34
.htm, .html	HTML files on your hard drive or the web
.doc, .wpd	Microsoft Word or WordPerfect documents
.pps	Microsoft PowerPoint shows
.bmp, .jpg, .gif	Images
.wav, .mp3	Sounds
.avi, .mpg	Videos
.exe	Programs (including old DOS, Windows 3.1, Windows 95, and Windows XP)
.csv	DirectRT Input file **

Purpose

The *File Name* field simply tells MediaLab which files to present in the different experimental conditions. See Questionnaire item types 43 for more details on specific file types.

Hints

Double-click on the *File* field to browse your hard drive and select files automatically. Press *Enter* to manually edit a file path.

To be safe, you can always specify the complete file path (e.g., *c:\experiments\myexp \myimage.bmp*). If you specify a complete path, remember to make sure the path is correct if you load the experiment files onto other computers.

Advanced Hints

If the file is located in the same folder as your experiment file, then you can simply enter the name of the file (e.g., *myimage.bmp*). If the file is located in a subfolder that is located in the same folder as your experiment files, then you can enter the name of the subfolder followed by the name of the file (e.g., *images\myimage.bmp*). Otherwise, you need to specify the full path and name of the file you want to present (e.g., *c:\mypictures \myimage.bmp*).

The advantage of placing the file in the experiment directory (or a subfolder) is that the experiment folder can then be moved to a different place and you won't have to worry about checking path names (e.g., *c:*\..., *d:*\..., etc.)

If the file is not in the same directory as the experiment, and it's not in a subfolder, then you must specify the complete path of the file, e.g., *c:\pictures\myimage.bmp*

Files may also be located on another computer on your local network. MediaLab can display files located on any computer on your network as long as the drive has been "mapped" on the system running MediaLab. For example, a hard drive on another computer may be mapped on your system as "h:\" or "s:\" etc. If you are new to mapping, ask your network administrator about "mapping" the drives of other machines on your network or refer to the instructions from Microsoft about mapping in Windows. Once the drive is mapped, you can refer to files on that drive just as you would local files (e.g., *h:\myfiles\myfile.bmp*). See also Running Experiments Over a Network 91 for details on how to run an entire experiment from another machine.

** It's generally not a good idea to place two DirectRT sessions directly back to back in your experiment because each DirectRT session requires a few seconds for one session to end and the next to begin. Instead, try separating the sessions with a simple instruction screen of some type.

4.6 BackGround

QuickInfo

Display an image in the background as the current file is presented.

Value

Any image file with a .bmp, .jpg, or .gif extension

Purpose

If you would like an image displayed in the background, simply type in or select the name of the image file in this field.

Advanced Hints

You can specify the location of the background image in the same way as you do for the primary image using parameter 31 values. This can be useful if you want to present two images (one as the main, and one as the background). To specify the location of the background image, place values for the top and left parameters 32 in parentheses after the file name. For example typing *myimage.bmp (t50,l200)* will place the background image 50 pixels from the top and 200 pixel from the left.

If you want to present images with Word or HTML documents, you can insert them directly into the Word or HTML documents.

4.7 BackSound

QuickInfo

If you would like a sound played in the background while another file type is presented then simply type or place in the name of a compatible sound file (in this field. Play a wave or mpeg (mp3) sound file in the background when the current file is presented.

Value

Any sound file with a .wav or .mp3 extension

Purpose

Backsounds are sounds files that play while another file is being presented. Keep in mind that MediaLab will play backsound files to their completion. This allows for a backsound to play through multiple items if you care to structure things that way. To have MediaLab wait for the duration of the sound file and then automatically proceed, use the delay parameter for the item and specify the length of the sound clip in seconds, e.g., for a 30 second sound clip, use (d30). You can find out the length of a sound clip by right clicking on the file in Windows Explorer and selecting Properties.

.

4.8 BackVideo

QuickInfo

Display a video in the background as the current file is presented.

Value

MediaLab will play most video files that are supported by the Windows Media Player.

Purpose

If you would like a video played in the background simply type in or select the name of

the video file in this field.

Advanced Hints

By default, the backvideo will be centered on the screen and play at its encoded size. To specify the location of the backvideo, use the top and left parameters 32 in parentheses after the file name. For example, *myvideo.bmp (t50,l200)* will place the backvideo 50 pixels from the top and 200 pixel from the left. To specify the size of the backvideo, use the width parameter 32 (the height will automatically be adjusted proportionally). For example, *myvideo.bmp (w640)* will play the video at a resolution of 640x480 pixels. A number of special width values are also available (e.g., quarter screen, half screen, full screen).

4.9 Randomize Within Groups

QuickInfo

Randomly orders and presents all files sharing the same RWG value.

Value

Any positive integer value

Purpose

To randomly order and present certain files in your experiment, assign the files you want to have randomized with the same number in this field. For example, if you have five files you want randomized, assign them each an RWG value of *1*. The five files will be randomly presented in the five positions they occupy in the experiment. All remaining files will be presented in their normal positions as long as their RWG value is either *0* or *none*. You can independently randomize another group of items by assigning them all a *2*, another group with *3*s and so forth.

Example

In one condition, you want to present 1) Word document, 2) five images randomly ordered, 3) five sounds randomly ordered, and 4) a final Word document. Assign an RWG value of "0" or *none* to the Word documents to keep them where they are, a "1" to the five images, and "2" to the five sounds. That's it. To randomize the order that groups of files are presented (e.g., whether the images or the sounds are presented first),see Randomize Between Groups 30 .

Hints

Use of the RWG field in the experiment files randomizes the order of questionnaires and related files, whereas using the RWG field in the questionnaire files 64 randomizes the presentation of items that generate the actual data. Thus, use this functionality when you want to randomize the higher-level elements of your experiment, the questionnaires themselves.

4.10 Randomize Between Groups

QuickInfo

All files sharing the same RBG value define a group of questionnaires or files to be kept together. If multiple groups are defined, then MediaLab will present them in a random order.

Value

Any positive integer value

Purpose

You may wish to randomly order and present entire groups of files in a given condition (e.g., randomizing whether personality questionnaires or demographic questionnaires come first) . To do this, assign the same RBG value to all of the items you want to keep together. For example, if you have three groups of ten files in one condition, and you want to randomize the order in which these 3 sets of files are presented, then assign each group a different number and make sure all the files within each group share the same number for that group. The files within each group will all stay together, but the groups themselves will be randomly ordered. All files assigned a *0* or *none* will not move. Note that all files within a group must be sequential in the given condition.

Example

In one condition, you want to present a Word document, five randomly ordered sets of five images, and a final Word document. Assign a value of "0" or *none* to the Word documents to keep them where they are, a "1" to the images in the first set, a "2" to the images in the second set, and so forth. That's it. To randomize the order that the files are presented within each group, see Randomize Within Groups 29 .

Hints

Use of the RBG field in the experiment files randomizes the order of groups of questionnaires and related files, whereas using the RBG field 64 Randomize Between Groups 65 in the questionnaire files 64 randomizes the presentation of groups items that generate the actual data. Thus, use this functionality when you want to randomize the groups higher-level elements of your experiment, the questionnaires themselves.

4.11 Parameters

QuickInfo

Parameters supply additional information and functionality for the presentation of certain files. In the experiment Editor, all parameters are optional but offer the user some nice flexibility. Parameters can be given in any order, must be separated by commas, and the entire set must be enclosed in a single set of parentheses. Note that parameters also can be set for background 27, backsound 28, and backvideo 28 files following the same rules except that their parameters are entered after the filename rather than in the parameters field.

Parameters

$	quickstyle 31
c	command line arguments 31
d	duration for timed items 33
l	left position 32
t	top position 32
w	width 32

Parameters for Different File Types

Questionnaire 43	duration (+/-), $quickstyle
Executable	command line arguments
HTML	duration
Image File	top, left, duration
Movie	top, left, width, duration
PowerPoint Show	none
Sound File	duration
Word Document	duration

Details

$quickstyle (optional)

To apply custom color and font settings to a questionnaire, you can create a *QuickStyle* 84 file from the *Preferences* menu. To apply it to a particular questionnaire file you can enter the name of the QuickStyle file precede by a $. E.g., *($mystyle)*.

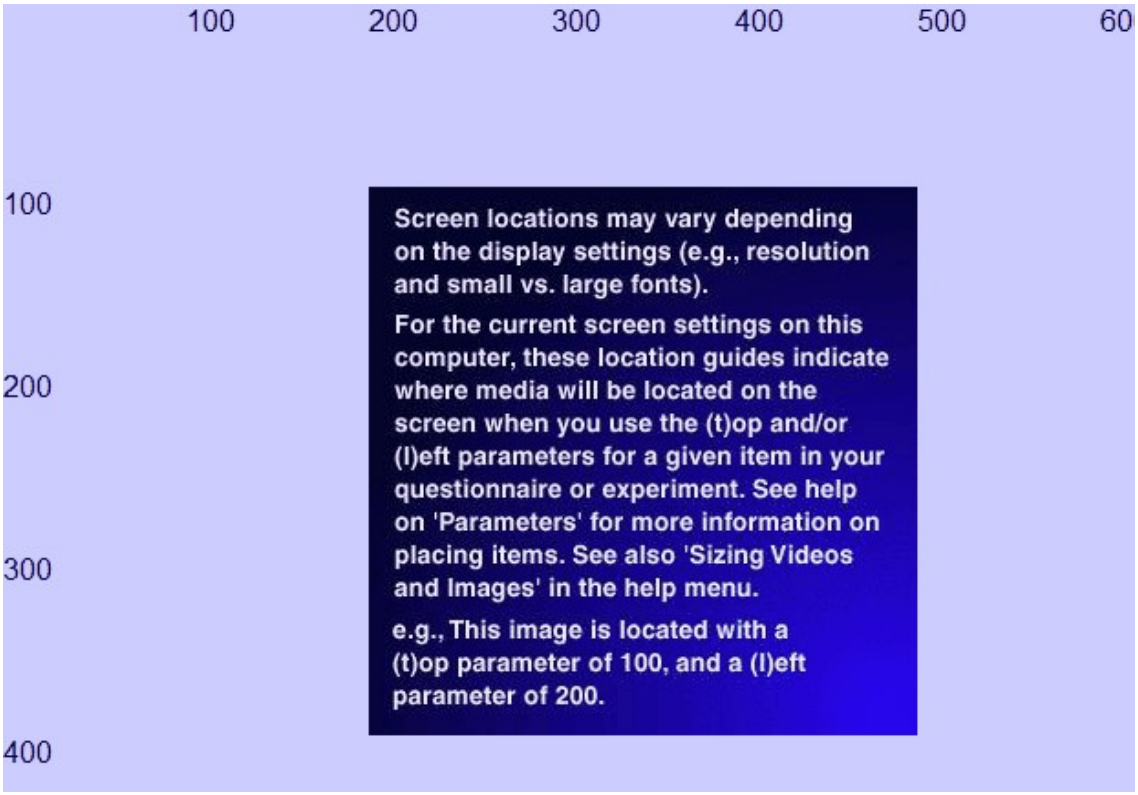
command-line arguments (optional)

For executables that allow for command line arguments, you can specify the arguments you want to send with the c parameter. Just type in the arguments preceded by a "c" (*cmyargument*). Follow any arguments with *max* and the application will be run in a

maximized window, e.g., *(cmax)*, or *(cmyarguments max)*

left position, top position (optional)

These parameters allow you to place an image or video exactly where you want it by specifying the top left corner of its position relative to the top left corner of the MediaLab window. If you wish to see exactly where the top left corner of your image will be placed, you can find the following guide by clicking on Help > Show Location Points when you first open MediaLab on your computer:



This window shows you exactly where the top left pixel of your image will be placed. The top and left pixel values are marked every hunderd pixels along the top and left borders, respectively.

The unit of measurement is pixels so placement will depend on what resolution you are running on your computer. MediaLab attempts to scale most screens so that the same scale is applicable no matter what resolution you are running. A left value of 400 and a top value 300 would be entered as (l400,t300).

Note: As of v2004, you can express ANY location and size parameters as a percentage of the screen. E.g., (w.3,h.25) would be a rectangle 30 percent of the screen's width and 25 percent of its height. To use percentages, simply use a value less than 1 and greater than 0.

width (optional)

By default, videos play in their original size. However, you can set the exact size of the video using the width parameter. The height will automatically be set in proportion to the width you specify. For example, if you specify *(w200)* for a video that is normally 320x240 it will appear as 200x150. You can also use the following shortcuts rather than specifying an exact width (e.g., *w-1*, *w2*, etc.)

- none or w0 = Default Size

- w-1 = Full Screen
- w-2 = Half Screen
- w-3 = One Fourth Screen
- w-4 = One Sixteenth Screen
- w1 = Double Size
- w2 = Half Size
- w10 and higher = specified width & proportional height

Note: As of v2004, you can express ANY location and size parameters as a percentage of the screen. E.g., (w.3,h.25) would be a rectangle 30 percent of the screen's width and 25 percent of its height. To use percentages, simply use a value less than 1 and greater than 0.

duration for timed items (optional)

To display one of the files in the experiment file for a predetermined amount of time, you can use the *d* parameter followed by the number of seconds you want that file to be activated. Once the amount of time in the *d* parameter has passed, the experiment file will automatically proceed to the next file. For example, if you wanted a questionnaire file activated for 30 seconds and then have the experiment file automatically proceed to the next file in it, you would specify *(d30)* for that first questionnaire file.

Questionnaire files also offer a special option you can use to create a special countdown clock to tell subjects how much time they have left to complete the questionnaire. On any questionnaire file, if you use a negative duration value (e.g., d-300) then a countdown clock will appear at the start of the questionnaire, showing how many second are left before the questionnaire will automatically end. The clock will start with the absolute value of the duration you specify (e.g., d-300 will put 300 seconds on the clock). If you are using a countdown timer for a questionnaire, try to avoid using duration parameters on items *within* that questionnaire that could conflict with your countdown (e.g., using the same number of seconds in the *d* parameters for the questionnaire in the experiment file and an item in that same questionnaire file). Note that you can place a single item in a questionnaire in order to get the countdown timer with a single item.

Examples

Image. To set the top left corner of an image to the center of a 640x480 resolution screen: *(l320,t240)* or *(l.5,t.5)*

Movie. To play a video file at full screen on a 800x600 resolution screen: *(w800)* or *(w-1)*.

Word Document. To display a Word document for 60 seconds rather than allowing the subject to press a key to continue: *(d60)*

Background. For a BackGround image to be displayed for 60 seconds with the top left corner located at the top left corner of the MediaLab window, specify the background file as: *myimage.bmp (t1,l1,d60)*

Key Concepts 34

Overview Window 35

Item Details

BackGround 39
BackSound 40
BackVideo 41
Filename 41
GoBack 42
Item Type 43
Item Wording/File 53
Labels 55
Name 56
Omit Conditions 57
Parameters 57
Position 64
Randomize Between Groups 65
Randomize Within Groups 64
SkipTo 66
Text Labels 67
Write Data as Text 69
Write Reaction Time 69
Write To Data File 68

Item Types

Custom 43
DirectRT 43
Essay 44
Executable 44
Fill-in-the-Blank 44
HTML 45
Image File 46
Inquisit 47
Instructions 47
Movie 47
Multiple Response 47
On-Line Rating 48
PowerPoint Show 48
Ranking 49
Recall List 49
Scale Response 50
Sound File 50
Thought Listing 51
Thought Rating 51
Voice Response 51
WebTracker 52
Word or WordPerfect Document 52

5.1 Key Concepts

It is recommended to first read key concepts of Experiment Files 20 before reading this section.

Questionnaire files are the workhorse of MediaLab. They are used to present and arrange the individual items of your experiment. Remember how we said earlier in the Experiment Files: Key Concepts 20 section that Questionnaire files were like jigsaw puzzle pieces? In the experiment file, you identify the files to be presented in your various conditions; this is where you can work with the individual items of those files.

The key concept for creating questionnaires is very similar to that of creating experiments. You just need to tell MediaLab what you want to present, and the order in which you want it to happen. You can use the Experiment Editor 11 to create, view and edit our questionnaire files. The primary difference is that in questionnaires you also ask

questions. This is why in questionnaires we use the term *items* rather than *files* to describe what is being presented. In contrast when talking about experiments, we use the term *files* because that is all that experiments can present.

Just like experiment files, questionnaire files can present various media images, videos, sounds, documents, html files, executable (.exe) programs, or PowerPoint shows. Unlike experiment files, MediaLab questionnaires use a variety of items 43. MediaLab's multi-media flexibility also allows you in most cases to combine items and media, presenting them simultaneously as part of a questionnaire file.

Another key concept to get with questionnaires is that they are self-contained. Once a questionnaire file has been created, it can be copied and simply dropped into any other experiment. This makes creating new experiments extremely easy when certain components are re-used (e.g., a questionnaire that measures a personality trait, or a questionnaire that contains a mood inducing video followed with manipulation checks, etc.). You can also run a questionnaire file on its own 9, but experiment files need other flies like questionnaires to run.

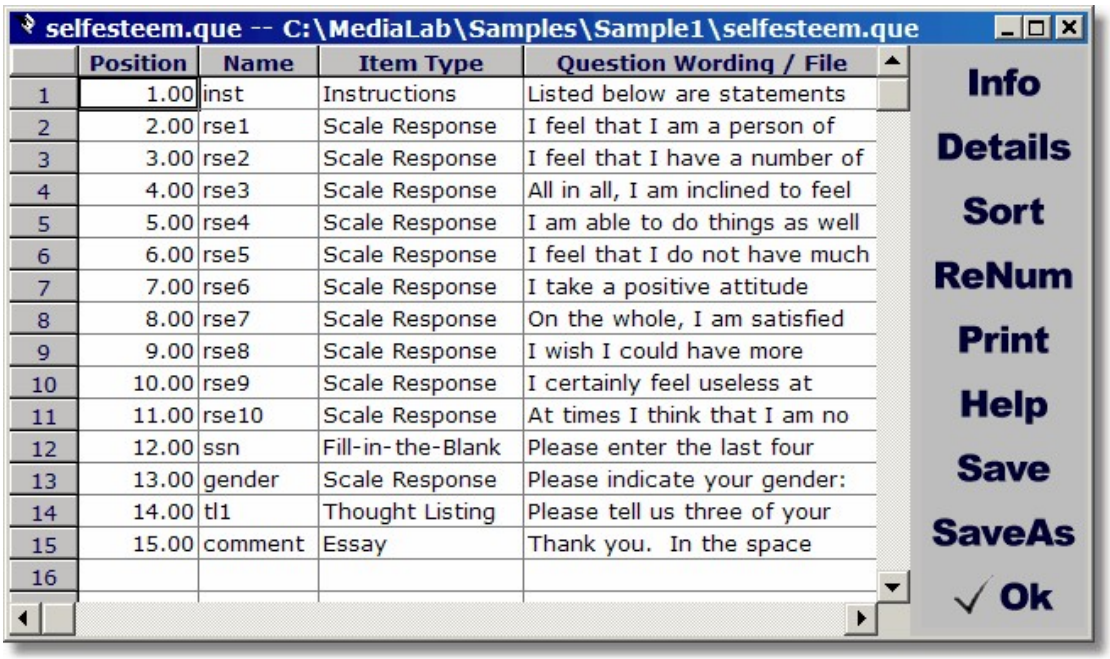
Remember that MediaLab help is context sensitive. When you are creating a questionnaire and you want to know what the purpose of a particular field is, just place your cursor in the field and then hit F1. The help files have been designed to take you right to the relevant information.

5.2 Overview Window

QuickInfo

In the overview editor, you can scroll through the basic information for each item in your questionnaire file. In this mode, you can see and define which items are presented to participants and the order in which they're to occur.

Click on a specific area for help on that topic:



Hints

Double-click on any position or name field, and the details will be displayed for that item. When your are editing in the overview window, double-clicking will allow you to edit any field or will pull open a file search box if a file is required. If you want to edit a field that requires a filename but do not want to open a file search dialog, then simply highlight the field and press enter to edit the field directly.

Info

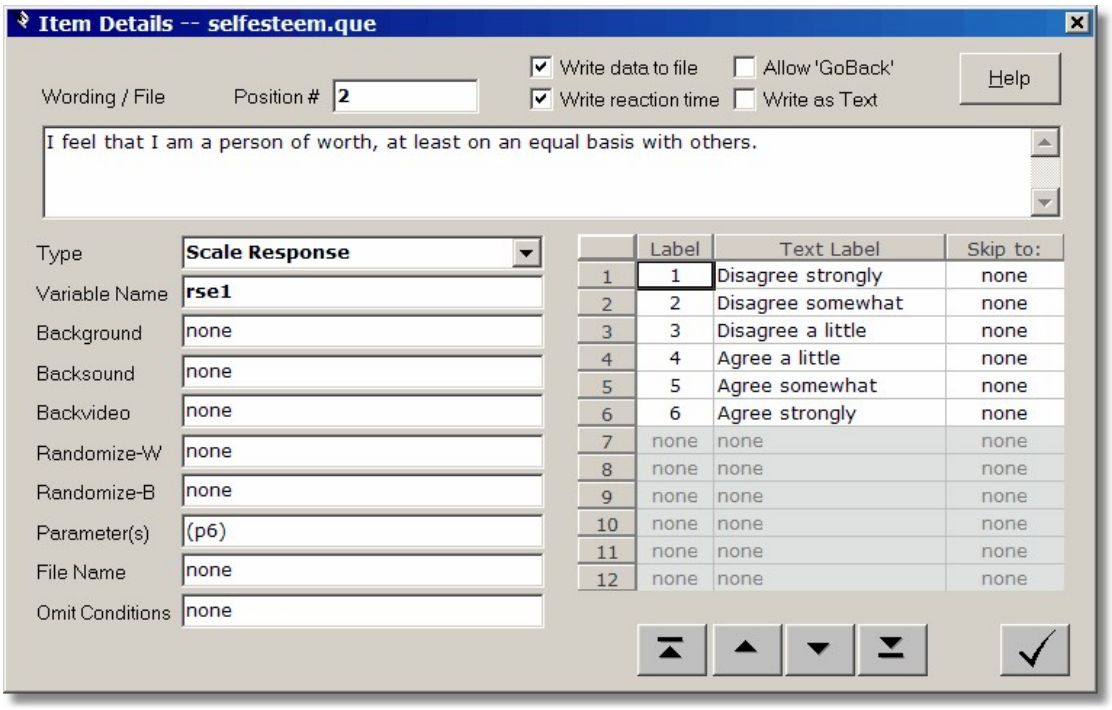
Allows you to save summary information and detailed notes regarding your questionnaire.

The title and comments are for the Editor's reference only. The comments field has a very large capacity if you wish to exploit it.

Details

Allows you to set various options for the presentation of your questionnaire items.

Click on a specific area for help on that topic:



Sort

Sorts the items in the questions by their position values.

The *Sort* button makes life a little easier by automatically reordering all of the items in your questionnaire so that you need not worry about where you are placing any one item at any given time. When MediaLab sorts the questionnaire items, it will reorder them by their ordinal position. If you forget to do this, MediaLab will do it automatically when you exit. You can manually sort whenever and as often as you like by clicking the *Sort* button.

Hint

To add a new item between two existing items (e.g., at positions 2 and 3), assign it a position value between these values (e.g., 2.5) and then click the *Sort* button. After sorting, you can automatically renumber all of your items with sequential integers by clicking the *Renumber* button.

Renumber

Renumbers the position 64 values in the questionnaire with sequential integer values.

The renumber button cleans up your position numbering by replacing the values with sequential integers. The first item will get a position value of *1*, the second will get a *2* and so forth. This is not a necessary operation and is provided mainly for the purpose of aesthetics and to help you know that you have the correct number of items your questionnaire.

Hint

If your items are out of order, be sure to sort 37 before you renumber.

Print

See Experiment Editor, Printing 16

Help

Starts the interactive help system. You can search the documentation using the Contents, Index and Search tabs in the left-hand window. Also note that MediaLab takes advantage of context sensitive help such that pressing F1 at any time will often direct you immediately to help with what you are working on. A printable PDF copy is available in the MediaLab program folder, installed by default as the C:\MediaLab\Help folder.

Save

Saves the active file with its current name and location on your drive.

Save As

Saves the active file but allows you to specify a new file name and/or save location.

OK

Closes the *Details* window and returns to the *Overview* window.

Navigation

There are many ways to navigate your way around the questionnaire editor. The most useful methods to be aware of are:

- To jump from field to field, you can simply click on the field you want to edit, or
- You can also use the *Tab* key and the arrow keys to jump from one field to the next.

When your are editing in the *Overview* window, double-clicking will allow you to edit any field or will open a file search box if a file is required. If you want to edit a field that requires a filename but do not want to open a file search dialog, then simply highlight the field and press enter to edit the field directly.

All forms have a button with a check mark, like this: . This button closes whatever editor you are currently in. You will always be given the option to save any changes that you've made.

Details Editor

In the details editor, you will notice four buttons with arrows:

- The first key will take you to the details of the first file in your questionnaire file,
- The second key takes you to the item immediately preceding the one you are currently editing,
- The third key takes you to the item immediately after the one you are currently editing,

- and the fourth key takes you to the last item in your questionnaire file.

5.3 BackGround

QuickInfo

Display an image or HTML file (or URL) in the background as the current item is presented.

Value

Any image file with a .bmp, .jpg, or .gif extension, or any HTML or URL.

Purpose

In questionnaires, the item types that support background images are:

Essays 44

Fill-in-the-Blank 44

Image files 46

Instructions 47

Multiple Response 47

Recall-listing 49

Scale Response 50

Sound files 50

Thought-listing 51

If you would like an image, html file or web page to be displayed in the background simply type in the name of the file in this field. If you want to present an image or HTML file by itself, simply use the Image 46 or HTML 45 item type.

Important Note

If you want to display an HTML file or URL as a background image you MUST specify the size and location of the area in which you want it displayed. This is because the browser display area will cover and hide most other items that it overlaps so it's very important to specify the area in which you want it (see below under Advanced Hints). To do this, use the top (t), left (l) 60, height (h), and width (w) 63 parameters. For example, if you have some specially formatted instructions in an HTML file that you want to use for a scale response, you could you could enter "instructions.htm (t50,l50,w400,h150)" in the BackGround field for that item. This defines an area for the html to appear that will not cover the buttons and labels of the scale response.

Advanced Hints

You can specify the location of the background image in the same way as you do for a primary image using parameter 57 values. This can be useful if you want to present two images (one as the main, and one as the background), or to present an image at the

same time as a question item (e.g., to get judgments of the image). To specify the location of the background image, use the top and left parameters 60 in parentheses after the file name. For example, myimage.bmp (t50,l200) will place the background image 50 pixels from the top and 200 pixel from the left.

If you want to present images within html, Word or Word Perfect documents, you can simply insert them into the html, Word, or WordPerfect files themselves.

Inserting some html with pop-up links in your background can allow your participants to click a link on the question page to see a message, helpful hints, instructions, translations, or any other text elaboration.

5.4 BackSound

QuickInfo

Play a wave (.wav) or mpeg (mp3) sound file in the background when the current item is presented.

Value

Any sound file with a .wav or .mp3 extension

Purpose

In questionnaire files, all items support the *Backsounds* field. If the item is a movie or sound file that already has sound content, then the content of the *Backsounds* field will be superimposed. If you would like a sound played in the background as another item is presented, simply type in the name of the sound file in this field.

Hint

Keep in mind that MediaLab will play files in the *Backsounds* field their completion. This allows for files in the *Backsounds* field to play through multiple items if you care to structure things that way. To have MediaLab wait for the duration of the sound file and then automatically proceed, use the duration parameter 60 for the questionnaire item using the *Backsounds* field and specify the length of the sound clip in seconds. For example, type (d30) in the *Parameter(s)* field of the questionnaire item to play the file in the *Backsounds* field for 30 seconds. You can find out the length of a sound clip by right clicking on the file in Windows Explorer and selecting Properties.

Note about Sounds versus BackSounds:

For flexibility in playing sound files, they can be presented as a Sound item 50 or they can be presented using the *Backsounds* field in other items. Note that Sound items end automatically when the next item in the questionnaire file starts, even if the sound file has not yet completed. In contrast, files in *Backsounds* field continue until completion (overlapping subsequent items) or until another file in a different *Backsounds* field is encountered. Thus Sounds items and the *Backsounds* field in other items can be used strategically to accomplish different effects. Note that if you need to stop a file in the *Backsounds* field upon item completion, you can simply insert "silence.wav" from the Utilities folder in the MediaLab program files in the *Backsounds* field for the subsequent item. It is a short silent sound file that will cancel any currently playing BackSound.

5.5 BackVideo

QuickInfo

Display a video in the background as the current item is presented.

Value

By default, MediaLab will play any video file that is supported by the Windows Media Player. This typically includes .mpg, .avi, .asf, .wmv, .dat, and .mpv video formats.

Purpose

In questionnaire files, all item types except executables, DirectRT and Inquisit sessions support the *Backvideos* field. If you would like a video played in the background simply type in the name of the video file in the *Backvideos* field of the item.

Advanced Hints

By default, the file in the *Backvideos* field will be centered on the screen and play at its encoded size. To specify the location of the backvideo, use the top and left parameters in parentheses after the file name. For example, *myvideo.bmp (t50,l200)* will place the backvideo 50 pixels from the top and 200 pixel from the left. To specify the size of the file in the *Backvideos* field, use the width parameter in parentheses after the file name. For example, *myvideo.bmp (w300)* will play the video at a resolution of 300 by its proportional height. As with any parameters, you may combine multiple parameters in a single set of parentheses.

5.6 Filename

QuickInfo

For item types that use files or programs from outside of MediaLab, the Question Wording/File Name field identifies the location of the external files or programs.

Value

Certain item types in questionnaire files use particular file formats in their Question Wording/ File Name field:

Item Type	Supported File Format
DirectRT	Any DirectRT .csv input file
Executable	Any .exe program
HTML	Any .htm or .html document

Image	Any .bmp, .jpg, or .gif image file
Inquisit	Any Inquisit experiment file
Movie	Any .avi, .mpg, or .mov video file
PowerPoint	Any Microsoft PowerPoint .pps file
Sound	Any .wav or .mp3 sound file
Word Doc	Any Microsoft Word .doc file
WordPerfect	Any WordPerfect .wpd file

Double-click on the *File* field to browse your hard drive and select files instead of having to type the name. To be safe, you can always specify the complete file path (e.g., "c:\experiments\myexp\myimage.bmp"). If you specify a complete path, remember to make sure the path is the same if you load the experiment onto other computers or drives.

Advanced Hints

If the file is located in the same folder as your experiment file, then you can simply enter the name of the file (e.g., myimage.bmp). If the file is located in a subfolder that is located in the same folder as your experiment files, then you can enter the name of the subfolder followed by the name of the file (e.g., images\myimage.bmp). Otherwise, you need to specify the full path and name of the file you want to present (e.g., c:\mypictures\myimage.bmp). The advantage of placing the file in the experiment directory (or a subfolder) is that the experiment folder can then be moved to a different place and you won't have to worry about checking path names (e.g., c:\..., d:\..., etc.)

Do not place any parameter 57 values in the File Name field. Any relevant parameters should go in the *Parameter(s)* field. Parameters should follow the filename only for files in the *Backgrounds* 39, *Backsounds* 40, and *Backvideos* 41 fields.

If you specify a file name ending with .txt for a scale response or fill-in-the-blank item, the subject's response to the question will be written to that text file preceded by a ~. This is so that other programs you may want to run during the session have the option of reading and acting on this information. It was particularly intended for use with DirectRT so that responses provided by subjects could be used as stimuli in an embedded DirectRT session.

5.7 GoBack

QuickInfo

If checked, an option will be provided to participants to return to the previous questionnaire item.

Values

True (checked) or False (unchecked)

Purpose

In the details window of a questionnaire file, you can check the GoBack option for any questionnaire item. If you check the GoBack option for any given item, then that item will be provide participants with a "Go Back" button which can be clicked to return to the previous questionnaire item. This is especially useful if you are concerned about participants entering responses they might want to go back and change. However, be

aware that only the participants last response to an item is recorded in the data files; if an item is encountered for a second time after the "Go Back" button is used, any previous responses to that item are not recorded in the data.

If you check the GoBack option for a sequence of items, then participants will be able to back up through the questionnaire. Note though that the GoBack function is limited to the active questionnaire file and will not let participants return to a previous questionnaire file from that experiment file.

5.8 Item Type

Item Types

Custom 88
DirectRT 43
Essay 44
Executable 44
Fill-in-the-Blank 44
HTML 45
Image File 46
Inquisit Session 47
Instructions 47
Movie 47
Multiple Response 47
On-Line Rating 48
PowerPoint Show 48
Ranking 49
Recall List 49
Scale Response 50
Sound File 50
Thought Listing 51
Thought Rating 51
Voice Response 51
WebTracker 52
Word Document 52
WordPerfect Doc 52

The following are descriptions of the various item types that can be included in a MediaLab questionnaire. See also information on how to set parameters 57.

Custom

Can't find what you need? MediaLab now allows you to create your own custom item types using Visual Basic, Visual C++ or other development platforms using the MediaLab ActiveX control. You can reuse your custom item type in other experiments or share it with others. Data are integrated with MediaLab's regular data files. Limited only by your imagination (ok, and maybe a little programming ability :)) See *Creating Custom Item Types* 88 for more information.

DirectRT

To embed a DirectRT session within your questionnaire, simply select the *DirectRT* item type and enter the name of the input file in the File field. MediaLab will pass the current subject and condition values to DirectRT which will run the session and then pass control back to MediaLab when it's finished. Note: DirectRT takes a few moments to start up and shut down. It's generally not a good idea to place two DirectRT sessions directly back to back in your questionnaire. Instead, try separating the sessions with a simple Instruction item (e.g., that says *Click Continue to proceed*). See DirectRT for more details on how to obtain DirectRT and what you can do with it. Note that MediaLab can save certain responses for later use as stimuli in a DirectRT session. 106

Essay

An essay item allows subjects to enter a response of unlimited length. Subjects will be given your question wording at the top of the screen and a box below it in which to type their response. Participants can voluntarily end the Essay item using the *Esc*, which you can explain in your question wording if you like, or it can automatically end by setting a time limit with the optional duration parameter 60. Data for essay questions will automatically be written to a separate file named with the variable name you assign to the item.

By default, the essay response box below the question wording takes up the remainder of the screen. Optionally, you can specify the size of the essay response box using the height (h) and width (w) parameters 63 and the placement of the essay response box using the top (t) and left (l) parameters 60. This allows additional room for question wording or for background images/html.

Note about reaction times. By default, response times for fill-in-the-blank and essay questions taken at the time the subject *continues to the next item*. You can optionally get it for the *first key stroke* instead by using (k1) in the Parameter(s) field 60 of the Essay item.

Executable

In the case that MediaLab can not meet your needs for a particular experiment or questionnaire, MediaLab will run any other program for you (including DOS, Windows 3.1, and Windows 95/98/NT/2000/XP programs). Simply embed the program within your questionnaire by selecting the executable item type and specify the file name in the *Question Wording/File Name* field. Hint: MediaLab writes the current subject number and condition to a file called *CurrentSubjectInfo.txt* at the beginning of every session in case you want your custom program to access this information. The Visual Basic code to do this is:

```
Open "c:\medialab\currentsubjectinfo.txt" For Input As #1
Input #1, subject: subject = Mid(subject, 11, Len(subject) - 10)
Input #1, condition: condition = Mid(condition, 11, Len(condition) - 10)
Close #1
```

If your program uses command line arguments, you can specify them with the optional command line parameter 59. You can also run executable programs for all other item types in a questionnaire file by appending the letter c to the beginning the file name of the executable program and placing it in the *Parameter(s)* field of that item, e.g., *(p7,cmyprog.exe).* This is especially useful for programs that should run silently in the background as MediaLab will *not* attempt to keep the executable program visually in the foreground *unless* it is launched via the Executable item type.

Note that embedded executables can not integrate their data with MediaLab's. If this is something you need then this can be accomplished with a *Custom item* 88.

Fill-in-the-blank

Exactly as it sounds, fill-in-the-blank items provide the question wording and space for a short open-ended answer. Fill-in-the-blank responses become available for insertion within the question wording and response options 76 of subsequent items.

You can limit the response to a number by using the range (r) parameter 62. Both positive and negative values are permitted when using the range parameter. For example, to accept only a valid age, you might use (r16-125) which would require a response between 16 and 125. The user will be prompted to enter a number within that

range if they fail to do so.

You can also use the mask (m) parameter[61] to specify the number of characters allowed in the response and whether each is fixed or free. For example, the parameter (m*****) will provide five blanks. The parameter (m***ck) will offer three blanks followed by "ck". Fixing characters in this way allows subjects to offer partial responses such as in a word completion task.

You can also combine the range and mask parameters. For example, the parameters (m*********, r100000000-999999999) could be used to prompt the subject for a valid, nine-digit social security number.

In cases where you need more space for a fill-in-the-blank item, you can use the optional space (s) parameter[62] to tell MediaLab how many lines of text you want to allow. For example, a parameter of (s3) will provide three lines of text instead of one (the default). Note that if you specify more than 1 line, the Enter key will not send the subject to the next item but rather to the next line in the text box. The Escape key may be used as an alternative to the Enter key in such cases.

in the same way you would use them for an image or essay item, you also use the top (t) and left (l) parameters[60] to control the placement of the response box as well as the height (h) and width (w)[63] parameters to control the size of the response box .

Note about reaction times. By default, response times for fill-in-the-blank and essay questions taken at the time the subject *continues to the next item*. You can optionally get it for the *first key stroke* instead by using the parameter (k1)[60].

HTML

Displays an HTML file exactly as it would appear in Microsoft's Internet Explorer. MediaLab will open a full screen browser window to display the HTML or other URL that you specify. The HTML page will be fully active so that the subject can click on hyperlinks or other objects and browse to other pages that you make available. HTML files can be located either locally or on the internet. Simply choose the HTML item and then enter the location of the HTML file in the Question Wording/File Name field (e.g., c:\experiment\stim1.htm, or www.mysite.edu/stim1.htm). Both .htm and .html are acceptable suffixes. To limit browsing by time, use the duration parameter[60]; otherwise, the subject can click *Continue* to proceed. Any Java or ActiveX applications will be fully supported.

Optionally, you can have navigation buttons appear while HTML pages are displayed. You can display any or all of: back, forward, home (will return to the starting URL), refresh and search. To enable this option, simply create a text file in your experiment folder called "browser.txt" and copy the following into it, changing true to false for any buttons you do not want displayed.

```
back = true
forward = true
stop = true
home = true
refresh = true
search = true
```

Note that the HTML item is capable of displaying many other file types than just HTML. Essentially, MediaLab passes whatever file or URL you specify to the Internet Explorer Browser engine. So, if you can view it in Explorer, then you can present it using the HTML item type. So if you have a file of any type that you want to display in MediaLab, try specifying it as the filename in an HTML item (see also Word Document item below).

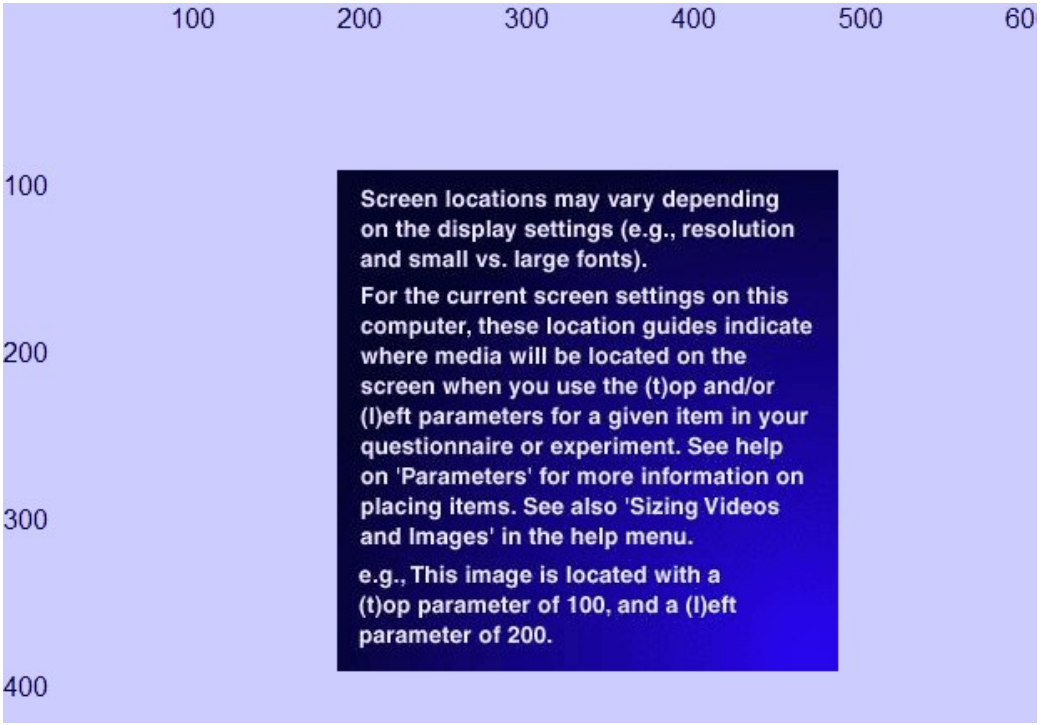
Caution: HTML pages that require input may have a subject pressing the *spacebar* as a part of that input. Since the *spacebar* is also used in MediaLab to continue with the next

item, this can cause a conflict. If this applicable to your HTML file, you can add a (k-1) parameter 60 to disable the *spacebar* as a means to proceed to next item.

Image File

Displays an image file. MediaLab supports BMP, GIF and JPG image formats. You can specify the location of the image on the screen using the optional top and left parameters 60. Hint: To display two images at a time you can select a background 39 image and place it accordingly, or you can insert multiple images in an HTML or Word file and then simply present that instead. To get ratings of an image, use a Scale Response and use the image as a background image.

Images by default appear in their actual dimensions. Optionally, you can use the height and width parameters to specify the image display size. Note that if you size an image using the height and width parameters 63 you need to also specify the top and left parameters 60 otherwise it will appear in the top left corner of the screen by default. Note that size and location parameters vary slight depending on your display settings. MediaLab provides a guide for understanding these values in the Help menu with an option called "Show Location Points:



Advanced Hints: BMP files will load faster than JPG or GIF formats because they do not have to first decompress. Also, 256 (8-bit) color images will load much faster than images encoded with thousands or millions of colors. Most image editing programs will allow you to convert images to 256 colors without a noticeable degradation in quality. If you do see a noticeable difference, try another image editor (e.g., IrfanView is very good and can be downloaded from http://www.irfanview.com). All MediaLab images (e.g., the blue screen, the MediaLab rats, the opening splash screen, etc. are 256 color images. The reason the display should be set to thousands or millions of colors is so that multiple 256-color images can be displayed simultaneously.

Inquisit

To embed an Inquisit session within your questionnaire you must have Inquisit build 1.25 (except build 1.32) or later installed and operational (see www.millisecond.com). Simply select the Inquisit item type and enter the name of the Inquisit experiment file in the File field. MediaLab will pass the current subject values to Inquisit which will run the session and then pass control back to MediaLab when it's finished.

Instructions

For many instructions, you may not want to go to the effort of creating a Word or HTML document. In many cases, simple text will do it. In such cases, choose the Instructions item and provide the text you want displayed in the *Question Wording/File Name* field.

Instructions items have a number of options to control their appearance. You can control the positioning of instruction text using the top and left parameters. You can also embed images, html, sound and video in an instruction item by using the *BackGround*, *BackSound*, or *BackVideo* fields on any *Instructions* screen.

Instruction items also allow you to specify the onset parameter like other items to create a delay before the text appears. Note that the response time (if you choose to record it) will include the onset value.

To control the amount of time that instructions are presented, you can use the duration parameter as you would with other item types. Instruction items also offer a unique option you can use to create a minimum exposure time. On any instruction item, if you use a *negative* duration value (e.g., d-10) then the *Continue* button will not appear until that many seconds have passed. You can use this option to ensure that subjects are exposed to some information for a minimum amount of time, after which they may proceed when ready.

By using the x parameter, you can require that a password be entered to continue beyond the instruction screen. This is a handy way to prevent subjects from continuing after a certain point without experimenter intervention. Note that you can also set a password for the end of all experiments in the Preferences section of the toolbar of the main MediaLab menu.

Movie

Presents a video file. By default, MediaLab will play any video file that is supported by the Windows Media Player. This typically includes .mpg, .avi, .asf, .wmv, .dat, and .mpv video formats.

You can specify the location of the video on the screen using the optional top and left parameters. You can specify the size of the video using the width parameter (the height will automatically be set in proportion). Movies can also be shown simultaneous with most other items types by specifying a video file in the *Backvideo* field.

Multiple Response

The "check all that apply" multiple response format is the standard format to use when you want to offer closed ended response alternatives AND allow subjects to select one or more of the response alternatives. Use from 1 to 12 response options and provide a text label for each. Subjects can select and deselect multiple response options by hitting the appropriate key or clicking the appropriate button with the mouse. You can also have them use the function keys (F1 through F12) in order to respond when 10 or more alternatives are presented. MediaLab will create a unique variable for each option and record whether or not each item was selected (1) or not (0).

Multiple response items require the p parameter 61 to indicate how many response options to use, e.g., (p5) for five options. You can optionally add a range parameter 62 to specify a minimum and/or maximum number of responses to take. For example, (p5,r1-3) would require that the subject choose at least 1 and no more than 3 of the 5 options. To require a single response, you could specify r1-1. Multiple response items support the *BackGround* 39, *BackSound* 40, or *BackVideo* 41 fields.

You can also now include *<specify>* in the text label of any response option. This will not be displayed to the subject but will cause MediaLab to prompt subjects to enter a text response. This is handy in cases where you want to provide an alternative such as "other." For example, if you wanted the seventh option to be "other" and you wanted subjects to specify what "other" means, then enter other *<specify>* for the text label. The actual text response will be written to the data file instead of whether or not the item was selected. If you choose to use this feature, you should select the *Write As Text* option for the variable since a numeric variable in SPSS will not be able to store a text response.

Multiple Response items support all of the same optional parameters as Scale Responses 50 (below).

On-Line Rating

The on-line rating feature allows you to capture continuous response data from subjects over the course of an entire item (e.g., movie, sound file, timed image or document). Using either a joystick (default if one is connected) or the left and right arrow keys, subjects will be asked to give continuous rating feedback for the duration of the item *following* the on-line rating item. All you need to do is set the number of scale points for the rating scale and the text labels for the scale points (use none to leave a label blank).

When the subsequent item starts, MediaLab will present the online rating scale which subjects can move up or down by using the joystick or arrow keys. MediaLab will take 10 samples every second and average these to provide a value for that second. MediaLab will continue doing so for the duration of the item and then write these values to a separate data file which will be named after the item being rated. For example, if you have subjects perform an online rating of a movie item called movie1, then all the online data will be stored in a file called movie1_online.csv. This file can be opened in Excel or other spreadsheet application for viewing.

Note that there is also a viewer included in the *Utilities* folder called "OnLineViewer". You can open any online output file to view the data plotted as a continuous graph with this program. You can also select by subject, by condition and you can set the time interval you want to see plotted. At any time, you can also copy the plotted values and paste them into Excel or another spreadsheet for further manipulation or graphing. There is also a joystick calibration program located in the *Utilities* folder as well.

PowerPoint Show

Presents a Microsoft PowerPoint show. This is an ideal way to present impressively formatted sequential display screens. Using PowerPoint, simply create a slide show and then save the file as a PowerPoint show (.pps format). Then, enter the filename in the Experiment editor in the position and condition(s) you want it to occur or enter it as an item within a questionnaire. PowerPoint shows do not require any parameters.

Hint: If you are new to PowerPoint, follow these steps to create your first slide show.

- Open PowerPoint and select File->New from the main menu. Choose the Presentations file tab and select a template (the Business Plan, Standard is very cool).
- Click on the text and replace it with whatever you like (e.g., 'Welcome to the experiment!"). Use the Page Down key to advance to the next slide and edit it as well.

- Add new slides with the Insert->New Slide command. Delete slides with the Edit->Delete Slide command.
- When you're happy with the show, select File->Save As from the main menu. In the window that appears, click on the Save As Type box and select PowerPoint Show (*.pps). Enter a name for the file.
- Drop it into your questionnaire or experiment file. You're done!
- Hint: Open PowerPoint and go to Tools->Options->View->SlideShow and click off all three options. This will eliminate the right clicking/popup issue as well as eliminate the "end of slide show" screen in case you're getting that as well.

Ranking

With the ranking item, you can give subjects up 12 items to rank in any order they choose. The items will appear as labels that subjects will be able to drag and drop into a special ranking area using the mouse. To design your ranking item, first enter a points parameter 61 to indicate how many items you want to have subjects rank. For example, to have 6 items ranked, enter *(p6)* in the *Parameter(s)* field. This will allow you to enter those 6 items in the *text labels* area. Enter any text you like. To have participants rank images, enter the name of each image file in a text label field. For examples of ranking (standard and with images), visit our Experiment Sharing area at support.empirisoft.com.

That is all that's necessary for a ranking item, but there are a number of formatting options. For example, you may choose to have an instruction window pop up at the start of the item. You can do so by entering text in the *Question Wording/File Name* field for the item. Subjects will be able to bring these instructions up again at any time by clicking the *Instructions* label at the bottom of the screen.

You can display labels at the top and bottom of the ranking area such as "Most Important" and "Least Important" so that subjects know which items go toward the top and which go toward the bottom. To do so enter these labels in the *Question Wording/File Name* field after the instructions (if any) like this:

=Most Important, Least Important

If you don't have any question wording then you can simply type the labels in the question wording field preceded by an equals sign.

By default, the ranking items will be as wide as they need to be to fit on one line. Long words and phrases will consequently be wider than short words and phrases. If you have entire sentences being ranked, you may find they are too wide. There are two ways to control the size of the items. First, you can specify a width parameter 63. This will force the items to be a constant width and the text will wrap to accommodate the width—i.e., becoming multi-lined. A second way you can control the size is by specifying a spacing parameter 62. E.g., to display 6 items with a fixed width of 250 pixels and a font size of 7, you would use parameter values of: (p6,w250,s7).

When saving the data, MediaLab will create a unique variable for each of your ranked items and save it's ordinal value in the ranking as its value. For example, if your ranking item's variable name were "rk1", then the fourth item in your list will be called rk1_04. If that forth item were ranked second, the value of rk1_04 would be 2. If you select the *Write Reaction Time* option for a ranking item, you will also get the first time each item was selected by the subject—these values will be recorded as typical RTs using the same variable name preceded with a t (e.g., trk1_04).

Recall List

Allows subjects to indicate what they can recall, one memory at a time. Operates in the same was as thought listings 51 (see below). Recall list items support the *BackGround* 39,

BackSound 40, or *BackVideo* 41 fields.

Scale Response

The scale response format is the standard format to use when you want to offer multiple closed ended response alternatives. Use from 1 to 12 response options and provide a text label for each. Subjects can select a response option by either using the number key that corresponds to the alternative (1 through 9) or you can have them use the function keys (F1 through F12) in order to respond when 10 or more alternatives are presented. Scale responses require the points parameter 61 to indicate how many scale points to use, e.g., (p5) for a five point scale. Scale responses support the *BackGround* 39, *BackSound* 40, or *BackVideo* 41 fields. The selected response also become available for insertion within the question wording of subsequent items 76.

By default, scale responses offer a single line for the text labels. In cases where you need more space because of long labels, you can use the spacing parameter 62. This allows you to create more space for labels with lots of text or if you're using an html file for your labels. For example, if you want to "double-space" your response options to create more space, use the parameter (s2), or (s3) to triple space, or (s1.5) to add a half space and so on. You can enter any value from 1 to 5 and decimals are ok. This will produce evenly spaced response options with as many extra lines per item as you need. Importantly you can do this on an item by item basis--the default of course is single space if you don't specify the space parameter.

By default, scale responses are presented vertically. They can be presented using a horizontal format by using the width parameter 63 to specify the width of the scale (see Help -> Show Location Points for help with sizing on your system). For example, (w.7) would make the scale span 70% of the screen width. Scales will be centered in this space.

With both horizontal and vertical scale formats, you can specify how far down the buttons appear by using the top parameter 60. This can help to create more space for extra question wording. Note that if you don't see your scale you might have moved it down too far (i.e., try a smaller top value).

To force a subject to answer with a correct response you can enter the item's variable name in *Skip to* field for any incorrect answers. For example, if a question's variable name is *q1* then simply enter *q1* in the *Skip to* field for any response options that are wrong. If one of these options is selected, the subject will be prompted to try again. Response times 69 will reflect the time of the first correct answer.

Finally, you may want a subject to specify their answer if, for example, you give the an "Other" option. Just include <specify> in the text label of the response option. This will not be displayed to the subject but will cause MediaLab to prompt subjects to enter a text response if they choose this option. For example, if you wanted the seventh option to be "other" and you wanted subjects to specify what "other" means, then enter other <specify> for the text label. The actual text response will be written to the data file instead of response number. If you choose to use this feature, you should select the *Write As Text* 69 option for the variable since a numeric variable in SPSS will not be able to store a text response.

Sound File

Plays a .WAV or .MP3 format sound file. For WAV files, larger sound files may take a few seconds or longer to load. Note that 8-bit mono sound files will load and play much faster than will 16 bit stereo sounds. In contrast, MP3 audio files are much smaller in size and will start to play immediately. Importantly, MP3 sound files are compressed with virtually no loss in sound quality! Sound files support the *BackGround* 39, *BackSound* 40 (which will play simultaneously), or *BackVideo* 41 fields.

Note about Sounds versus BackSounds

For flexibility in playing sound files, they can be presented as a Sound item or they can be presented as *BackSounds* to other items. Note that Sound items end automatically when the next item appears even if they have not yet completed. In contrast, *BackSounds* continue until completion (overlapping subsequent items) or until a file in another *BackSound* field of a different item is encountered. Thus Sounds and BackSound can be used strategically to accomplish different effects. Note that if you need to stop a *BackSound* upon item completion then you can simply insert "silence.wav" in the *Backsound field* for the subsequent item. It is a short silent sound file that will cancel any currently playing *BackSound*. Silence.wav can be found in the Utilities folder.

Thought Listing

The thought listing item is used when multiple open-ended responses are desired for a single question. For example, a common use for this item type is to expose subjects to information and then ask them to indicate the thoughts that occurred to them during exposure. Thought listing items require the n parameter 61 to indicate the number of responses you want listed, e.g., (n4) for four thoughts. You can also specify a time limit for the thought listing item using the duration parameter 60. Note also that the escape key will end the thought listing question. You can optionally inform subjects of this so that they can continue whenever they are done listing their thoughts.

Thought Rating

MediaLab allows you to have subjects rate their own open-ended responses from thought or recall lists on any dimension you like. The thought rating item works much like a scale response but feeds back to subjects their responses from an earlier thought or recall list question. To do this, you need to specify the variable name you gave to the thought or recall list from which MediaLab is to supply the responses. For example, say subjects listed their thoughts earlier in a variable you named *tlist*. To have subjects later rate these thoughts using a thought rating item (for example how positive or negative each thought was), you would specify (qtlist) in the *Parameter(s)* field your though rating item. Note that for thought rating items, you must also specify the number of scale points parameter the points parameter 61. Thought rating items support all of the same optional parameters as Scale Responses (above) as well as the *BackGround* 39, *BackSound* 40, or *BackVideo* 41 fields.

You can also include <specify> in the text label of any response option. This will not be displayed to the subject but will cause MediaLab to prompt subjects to enter a text response. This is handy in cases where you want to provide an alternative such as *other*. For example,iIf you wanted the seventh option to be *other* and you wanted subjects to specify what *other* means, then enter other <specify> for the text label of the response choice. The actual text response will be written to the data file instead of whether or not the item was selected. If you choose to use this feature, you should select the *Write As Text* 69 option for the variable since a numeric variable in SPSS will not be able to store a text response.

Voice Response

MediaLab allows you to record voice responses and optionally measure the reaction time of those responses. To save a voice response as a sound file, simply enter a name in the Question Wording/File Name field of the Voice Response item. It will automatically be saved by that name in a *Voice* subfolder within the main Data folder. If you specify *none* then no sound file will be saved. If you check *Write Reaction Time* 69 then you will get the latency of the response in milliseconds. You can check this option and specify *none* for the filename if you are interested only in the RT.

By default, MediaLab will record the voice response until the subject clicks *Continue*. You can optionally specify a duration parameter 60 to record the voice response for a fixed time. For example, (d5) would record a voice response for 5 seconds and then automatically continue to the next item.

Voice Response Setup

Recording voice responses and voice RTs requires that Microsoft's DirectX, Version 7 or later be installed on your system. You can determine what version (if any) you have by typing *dxdiag* from the *Run* command on the Windows Start Menu. DirectX can be downloaded from http://www.microsoft.com/en-us/download/details.aspx?id=35.

You will need to make sure you have a microphone connected to the microphone input connector on your sound card. You will also need to make sure that the microphone is active and able to record. Finally, you will need to tell MediaLab how sensitive it should be in detecting response times.

To make all this easier, MediaLab comes with a tool called SoundCheck in the *Utilities* folder. After making sure your microphone is plugged into your soundcard, click *Monitor* and speak into the microphone. If all is well, you should see the bar labeled *Microphone Level* respond immediately to your voice. If that works, then click *Stop* to end the monitoring. Now try a sample voice rt. Click *Test RT* and then speak into the microphone. SoundCheck should report the response time in milliseconds.

If you need to speak too loudly to set off the test, then try moving the sensitivity slider toward the left. If the test is set off too easily, then try moving the sensitivity slider toward the right. Try this until you find an RT sensitivity that works well for you. When you are happy with the sensitivity, click *Save*. This will make the new sensitivity level available to MediaLab when you run a session with voice responses. It is advisable to run a few test RTs on any machine before running a MediaLab session with voice responses for the first time. Note: Some users have reported that optimal performance for Voice RTs requires restarting MediaLab between sessions. While we work on figuring out why, we'll simply pass on this recommendation.

Web Tracker

The HTML item type allows you to present HTML and other "browser-compatible" files. The WebTracker item extends this functionality by tracking a subject's behavior while browsing through a sequence of HTML or other browser-compatible files. It stores each link that the subject follows as well as the time at which the link was followed and how long the subject stayed there before continuing to the next link. The item ends when the subject clicks the "Continue" button. These data are all written to a separate spreadsheet file (.csv) that will be named after the variable name you assigned to the WebTracker item and will be located in the experiment's data folder.

Word and WordPerfect Documents

Note: Some systems have a hard time displaying Word documents. If you have any trouble, simply save your documents in Word as HTML (using Save As) and then present them as HTML items instead.

Present nicely formatted text by inserting a Word or WordPerfect document. Since MediaLab presents these documents exactly as they appear when you see them in Word or WordPerfect, you can exploit all of the formatting power available in these applications. Basically, however it looks in Word or WordPerfect is how it's going to look in MediaLab. You can even embed images or other objects in your document and they will appear in MediaLab. Word and WordPerfect documents support *BackSound* 40, or *BackVideo* 41 fields. Each application must be installed on the system in order to take advantage of this functionality.

Some important considerations for using Word and WordPerfect documents with MediaLab

- Each screen you want to show must be a separate Word/WordPerfect document. Documents that will not fit on one screen can be broken up, saved as separate files and presented sequentially. If you need all of your document presented in one long sequential file that users can scroll through, consider saving your file as HTML and presenting it as an HTML item (see above)
- Document margins and text size will depend on your display resolution. Wider documents can be viewed when your monitor display is set at 800x600 than when it is set at 640x480 and so forth.
- Some documents may appear without regard to their top and left margins (i.e., the text will appear to the very left and very top of the screen). If you experience this, don't bother to adjust your margins. Instead, select the text and increase the paragraph indentation (on the menu, select *format* > *paragraph*.). For extra space at the top, insert hard returns. This solves the problem.
- Microsoft Word may warn you about macros when MediaLab attempts to present a Word file. You can choose to disable macros, or you can save your documents without macros. Either way, if this happens, you can tell Word not to warn you about this every time so that this window does not pop up every time MediaLab displays a document with macros in it.
- MediaLab uses a 2-second keyboard/mouse cancellation buffer for Word/WordPerfect documents to avoid accidental skipping of Word documents. This is done because Word documents may take a second or two to load and subjects may "continue" repeatedly while this is happening.

Note that Word Document item is also capable of displaying many other file types than just Word Documents. If you have a file of any type that you want to display in MediaLab, try specifying it as the filename for Word Document items (see also HTML items above). MediaLab will attempt to cooperate with the file's usual program to have it displayed correctly (e.g., an Adobe Acrobat file, an Excel Spreadsheet, a Flash Presentation, etc.). Of course this depends on how cooperative the other program is but it's definitely worth checking if you're curious.

See also

Appendix A: Unusual Task Specific Features 106 for a few functions that are task specific.

5.9 Item Wording/File

QuickInfo

Identifies the question wording for the questionnaire item type being presented.

Value – as Question Wording

Use Question Wording/ File Name field to create the question wording for the following item types:

Fill-in-the-blank

Instructions

Multiple Response

Recall Listing

Scale Response

Thought-listing

Thought-Rating

Hints – as Question Wording

Click *Details* for a more flexible editing space.

A very simple solution for more elaborate or formatted instructions is to simply insert them in a html, word or word perfect document and present that as its own item.

Advanced Hints – as Question Wording

If you want more formatting control over question wording you can insert HTML anywhere on any question item by specifying an HTML file or URL in the *BackGround* 39 field for the item. This way you can create whatever you like and present it in place of (or in addition to) the usual instruction text. Note that you must specify the area in the which the HTML should appear or it may hide other parts of the question item.

You can insert the value of a subject's prior response to any fill-in-the blank or scale response item by typing the appropriate variable name surrounded by < > within any question wording text. For example, if you had previously asked subjects about their race using a fill-in-the-blank question or a scale response and you had called this variable race, then you could insert the subjects response in a subsequent question wording like this: What percentage of your friends are <race>? For scale responses, the text label of the selected response is used.

Apply parameters to question wording

You can add parameters in the question wording itself specific to the question wording on any item that supports question wording. You can add top (t) and left (l) parameters 60 to define the questions location on the screen as well as width (w) and height (h) parameters 63 to define boundaries of the presentation box of the wording. To do so, use the key word "param" at the end of your question wording followed by your parameters in parentheses. For example, you could enter the following in the Questoin Wording/ File Name field:

Please answer the following questions. When you're done, feel proud you did a good thing. param(t200,l100,w200,h100).

This would cause the the question wording to appear 200 pixels down and 100 pixels to the right of the top left corner of the screen in a box three lines high and five words wide. You may have to try a few combinations of settings before you find one you like.

You can also add an "a" parameter to set the alignment of the text. By default it will be left aligned. For right alignment, add "a1". For center alignment, add "a2". e.g., "The quick brown fox jumped off the roof. param(a1)" will be right aligned. "The quick brown fox jumped off the roof. param(a2,t.5)" will center the sentence horizontally on the screen and the top of the text will be located 50% of the way down the screen.

5.10 Labels

QuickInfo

For scale response, multiple response, thought rating, and online rating item types, the buttons labels determine what is shown on the buttons when that item is displayed onscreen. The default labels are the numbers from 1 to 12, starting with the top response option. These values can be changed to whatever you want (e.g., A, B, C, -1, -2, -3, etc.).

When using the keyboard, these numbers correspond to what number key (1-9 for the first 9 options) and/or function key (F1-F12 for any of the 12 options) can be pressed to select a button. However, any button can be selected using a mouse to move the cursor to the desired button and left-clicking it.

Advanced Hints

You can modify the the size, format and location of response options and buttons using the *Text Label* 67 field to the right of the *Label* field.

On scale responses and thought ratings, the default value written to the data file is always the ordinal value of the response option. For example, if the first option is chosen then a *1* is written to the data file—no matter what is in the *Label* field for that button. For multiple response items, a button that is chosen and shows a check mark when the Continue button is pressed will be recorded as *1* while buttons that are not chosen are recorded as *0*.

For all of these item types you can specify an alternate value to record in the data file for any given response option by placing it in squiggly brackets {} after the text label like so:

	Label	Text Label	Skip to:
1	1	strongly disagree {-2}	none
2	2	disagree {-1}	none
3	3	neutral {0}	none
4	4	agree {1}	none
5	5	strongly agree {2}	none

Starting with the "strongly disagree" option and going down the list, these options will now be recorded as *-2, -1, 0, 1,* and *2*, respectively. Note that the value can be text or numeric although numeric is recommended. If you use text, make sure the *Write as Text* 69 option is checked for the item.

Input via Serial Port

As an alternative to keyboard input, you can receive input from external devices such as response boxes via the serial port. To do so you just need to create a file called "comport.txt" in your experiment folder and copy the following text into it:

```
comPort, baudRate, parity, dataBit, stopBit
1,19200,n,8,1
input code (1-255), response value (1-12)
1,1
2,2
3,3
4,4
5,5
6,6
```

You should not modify the first or third line. The second line represents the values of the variables listed on the first line. You can change any of these as necessary. The remaining lines tell MediaLab how to map incoming signals to response keys. On each line you can list a pair where the first value is the signal sent to the serial port (i.e., 1-255) and the second value is the response it should map on to (i.e., 1 to 12). In the example above, if MediaLab detects a "1" coming through the serial port, it will react as though the 1 or F1 key had just been pressed. See the *Serial Data* 107 section for more information.

5.11 Name

QuickInfo

Identifies the item name and provides a variable name for the data file.

Values

Any string, up to 60 characters. Keep in mind though that some statistical packages and spreadsheet applications may still have difficulty working with longer variable names. Avoid spaces, special characters and variable names that start with a number. Be sure to view and analyze some practice data before going too crazy with this feature.

Purpose

This field is the variable name. A unique name is required for every item in the questionnaire. This is the name that will be written to the SPSS and SAS input files. For items that collect multiple responses, numbers will be appended to the name. Item names also allow for skip patterns 66 and thought ratings 51 and for the insertion of subjects' responses into later question wordings 76.

Hints

Most data analysis programs react negatively to a few things that can be avoided here. For example:

- Avoid variable names that begin with a number.
- Avoid using the same variable name for different items.

Some other things to know:

- Reaction times are represented in you data input file by a 't' added to the corresponding variable names.
- Questions that gather multiple responses (e.g., a thought listing question named *thlist*) are represented by an incremental index appended to the variable name (e.g., *thlist_1*,

thlist_2, *thlist_3*, etc.). Consequently, to keep the output variable name to a maximum of 8 characters, keep in mind that 2 to 4 additional characters will be appended to the variable name (e.g., *_4*, *_12*, plus a *t* prefix for the reaction time variables if they're being recorded). If you always limit such variable names to a maximum of four characters you will always be safe.

5.12 Omit Conditions

QuickInfo

Omits the particular question in the conditions specified. Response is assigned a missing value.

Details

Although you may want to administer a given questionnaire in multiple conditions, you may want to skip certain items in some conditions. To skip the item, simply enter the condition(s) that are not to receive the item in the omit conditions field.

Example

Subjects watch a video about sexual harassment of women in the workplace by male co-workers. Afterwards they fill out a female sexual harassment questionnaire. Men are all assigned to condition *1* and women are assigned to condition *2*. The question "Have you ever been sexually harassed by a male co-worker?" is to be omitted for male subjects. Therefore, in the *Omit Conditions* field, there will be a *(1)*. The question "Have you ever sexually harassed a female co-worker" is to be omitted for female subjects, and so *(2)* is placed in the *Omit Conditions* field.

If more than one condition should not get the item, then separate the conditions with commas, e.g., *(1,2,6,cond3).*

5.13 Parameters

QuickInfo

This is about as difficult as MediaLab gets. Parameters supply additional information for the presentation of certain questionnaire items. Most are optional but a few are required for certain item types. Parameters can be given in any order, must be separated by commas, and the entire set must be enclosed in a single set of parentheses. Note that parameters also can be set for background, backsound, and backvideo files following the same rules except that their parameters are entered after the filename rather than in the *Parameter(s)* field.

Parameters

$	quickstyle 59
@	time stamp 59

c	command line arguments 59
d	duration for timed items 60
h	height 63
k	key 60
l	left position 60
m	mask 61
n	number of thoughts to take 61
o	question wording onset 61
p	number of scale points (1-12) 61
q	target question for thought rating 62
r	range 62
s	spacing 62
t	top position 60
ttl	send ttl signal 62
w	width 63
x	password required to continue 61

Parameters for Different Item Types – Required parameters are in bold CAPS

All Items 43	$quickstyle, @time stamp, ttlsignal, command, -key
Custom 43	none
DirectRT 43	none
Essay 44	duration, top, left, width, height, key
Executable 44	command line arguments
Fill-in-the-Blank 44	duration, top, left, width, height, mask, range, key
HTML 45	duration
Image File 46	top, left, width, height, duration
Inquisit Session 47	none

Instructions 47	duration, password, onset, top, left
Movie 47	top, left, width
Multiple Response 47	**P**oints, top, spacing, onset, range
On-Line Rating 48	**P**oints
PowerPoint Show 48	none
Ranking 49	**P**oints, spacing, width
Recall List 49	**N**umber, duration, onset
Scale Response 50	**P**oints, top, spacing, onset, width, left
Sound File 50	duration
Thought Listing 51	**N**umber, duration, onset, width, left
Thought Rating 51	**Q**uestion, **P**oints, top, spacing, onset, width, left
Voice Response 51	duration
WebTracker 52	none
Word Document 52	duration
WordPerfect Doc 52	duration

Details

Note: (optional) indicates the parameter is NEVER a required parameter of ANY item type.

$ quickstyle (optional)

To apply custom color and font settings to a questionnaire item, you can create a *QuickStyle* 84 file from the *Preferences* menu. To apply it to a questionnaire item you can enter the name of the *QuickStyle* file precede by a *$*, e.g., *($mystyle)*. The style will remain in effect until you apply different style file on a subsequent item or enter *$off* as a parameter for a subsequent item.

@ time stamp (optional)

By entering @ as a parameter for any item in a questionnaire, you can have MediaLab write a time stamp at the onset of that item in order to keep track of events in real time. When you do this, a time stamp will be appended to a file called "timestamps.csv" located in the *Data* folder. Each time stamp will include the Subject ID, the variable name, the time at which the item started (e.g., 12:33:22), the milliseconds elapsed from the beginning of the experiment and the number of milliseconds passed since the last timestamp. The timestamp function can be used with all item types.

command-line arguments (optional)

For executables that allow for command line arguments, you can specify the arguments you want to send with the c parameter. Just type in the arguments preceded by a *c* (e.g., *c/myargument*). Follow any arguments with max and the application will be run in a maximized window (e.g., *cmax*, or *c/myarguments max*).

For other item types, an executable file may be launched at the start of the item by adding a "c" parameter followed by the file name. For example, *(p7,cmyprog.exe)* in the

Parameter(s) field would cause *myprog.exe* to launch at the start of a 7pt scale response. This is especially useful for programs that should run silently in the background as MediaLab will *not* attempt to keep the program visually in the foreground *unless* it is launched via the regular Executable item 44 type.

duration for timed items (optional)

The default for many item types (e.g., instructions, essays, Word documents, fill-in-the-blank, thought and recall lists, image files) is to click on Continue to proceed. Optionally, you can set a duration parameter *d* followed by a number to display these items for a set amount of time and then automatically proceed to the next available item. For example, if you wanted an instruction or an image to be displayed for 30 seconds and then automatically continue, you would specify *(d30)*. An integer value greater than or equal to 1 is required.

Instruction items also offer a special option you can use to create a minimum exposure time. On any instruction item, if you use a negative duration value (e.g., *d-10*) then the 'Continue' button will not appear until that many seconds have passed. You can use this option to ensure that subjects are exposed to some information for a minimum amount of time, after which they may proceed when ready. Remember that you can always embed images & html, sound and video in an instruction item by using *BackGround* 39, *BackSound* 40, or *BackVideo* 41 fields, respectively.

key

Open Ended Response Times: By default, response times for fill-in-the-blank and essay questions are taken at the time the subject *continues to the next item*. You can optionally get it for the *first key stroke* instead by using the parameter (*k1*).

Disable the Spacebar: HTML pages that require input may have a subject pressing the *space bar* as a part of that input. Since the *space bar* is also used in MediaLab to continue with the next item, this can cause a conflict. If this applicable to your HTML file, you can add a *(k-1)* parameter to disable the *space bar* as a means to proceed to next item.

Track Key Presses: When used with an essay item, this parameter allows you to record every time a given key is pressed over a given time period. Simply add the key parameter to indicate that you want to track a key press response over the duration of the essay. You need to tell MediaLab what key you want by giving it a code (e.g., *k57* will have MediaLab track the *9* key). There is a utility included in the *Utilities* folder called *keycodes.exe*. It will quickly tell you the code for any key on the keyboard including function and non-character keys. Just double click the keycodes.exe program and hit any key to get its code. These codes are *NOT* the same as the more extensive set of DirectRT key codes!

Here's an example. You want to track every time the subject hits the *9* key over a 30 second interval. Run *keypress.exe* and hit the *9* key. You will see that its code is 57. Using an essay item, enter the parameters (d30,k57). This means the item will be displayed for 30 seconds and track every time the user hits the *9* key. Medialab will write the key tracking data to a file called the same thing as the essay variable name file but with "keypress.csv" added to it. You can open the data file in Excel or similar spreadsheet program. It will indicate the *subject* and *condition* IDs and give you the time in milliseconds from the start of the essay for each time the key was hit.

left position, top position (optional)

These parameters allow you to place an image, text or video exactly where you want it by specifying the top left corner of its position relative to the top left corner of the MediaLab window, e.g., *(t150,l200)*. You can also express location and size parameters as a percentage of the screen. For example, *(l.3,t.25)* would be a point 30 percent across the screen from the left and 25 percent down. To use percentages, simply use a value less than 1 and greater than 0. To determine values appropriate for your display, select

"Showing Location Points" under the main MediaLab Help menu.

The left and top parameters can also specify the left and top position of the essay, fill-in-the-blank, and thought listing text boxes as well as the text presented in instruction items. You can also use the top parameter to specify the location of the first button when using scale responses, multiple responses and thought ratings (e.g., when you need more room for question wording).

mask (optional)

You can use the mask parameter to specify the number of characters allowed in a fill-in-the-blank item as well as whether each is fixed or free. For example, the parameter *(m*****)* will provide five blanks that are free to be filled by any character. The parameter *(m***ck)* will offer three free blanks preceeded by "m" and followed by "ck". Fixing characters in this way allows subjects to offer partial responses such as in a word completion task.

You can also combine the mask and range 62 parameters. For example, the parameters *(m*********, r100000000-999999999)* could be used to prompt the subject for a valid, nine-digit social security number.

number of scale points (1-12)

For scale responses, multiple responses, thought ratings, and on-line rating items you need to specify the number of scale points you would like MediaLab to display. When you enter this parameter, MediaLab will then allow you to enter the labels for the number of points you specify. You can have from 1 to 12 points. For example, to specify a 5-point scale you would use *(p5)*. Note that when MediaLab runs your experiment, the number keys (1-9) will work only for a scale of up to 9 points. However, MediaLab will also take input from the function keys (F1 to F12) so that you can conveniently collect data from a 12 point scale and still assess reaction time. This does not apply to on-line rating items because input is taken through the arrow keys or joystick.

onset of question wording (optional)

For items that take the onset parameter, you can delay the appearance of the question wording for a specified number of milliseconds. For example, a 7pt scale response with the parameters *(p7,o750)* will reveal the options immediately, but the question wording will not appear for 750ms.

maximum number of thoughts or recall items to take

For thought listing and recall list itmes, you need to specify the number of separate responses you would like to record. For example, if you wanted the subject to list four thoughts in a thought listing item, then you would specify *(n4)*. If you wanted to them to list up to 4, you can set a time limit using the duration parameter 60.

password (x) (optional)

Sometimes you may want the subjects to wait for you to allow them to continue to the next phase of the experiment. You may also want to prevent the subject from seeing the MediaLab screen when the experiment is finished. To do this, insert an instruction item and use the *x* parameter followed by the password you want to use. For example, if you wanted to use the password 'monkey', then enter *(xmonkey)* as a parameter value for the instruction item. Presumably, the experimenter will have to enter this password or tell subjects what it is at the appropriate time. The characters will be masked (e.g., ******), so that subjects will not see the password as it is entered. Pressing *<Enter>* after entering the correct password will cause MediaLab to continue.

range (optional)

You can now limit a fill-in-the-blank item 44 response to a numerical range by using the range parameter. For example, to accept only a valid age, you might use *(r16-125)* which would require a response between 16 and 125. The user will be prompted to enter a number within that range if they fail to do so. You can optionally combine this with the mask parameter 61 to provide the appropriate number of spaces for the number. You can also add a range parameter 62 to multiple response items 47 to specify a minimum and/or maximum number of responses to take. For example, *(p5,r1-3)* would require that the subject choose at least 1 and no more than 3 of the 5 options.

spacing (optional)

For items that use response buttons (i.e., scale responses, multiple responses and thought ratings) you can use the spacing parameter to create more space for options with lots of text or if you're using an html file for your labels. For example, if you want to "double-space" your response options, use the parameter *(s2)*, or *(s3)* to triple space, or *(s1.5)* to add a half space and so on. Values from 1 to 5, including decimals, are permitted. This will produce evenly spaced response options with as many extra lines per item as you need. Importantly you can do this on an item by item basis--the default of course is single space if you don't specify the space parameter.

For fill-in-the-blank items 44, you can use the Space parameter to tell MediaLab how many lines of text you want to allow. For example, a parameter of *(s3)* will provide three lines of text instead of one (the default). Note that if you specify more than 1 line, the Enter key will not send the subject to the next item but rather to the next line in the text box. The *Escape* key may be used as an alternative to the Enter key in such cases.

target question for thought rating (q)

MediaLab allows you to have subjects rate their own open-ended responses from thought or recall list items on any dimension you like. The thought rating item works much like a scale response but feeds back to subjects their responses from an earlier thought or recall list item. To do this, you need to specify the variable name you gave to the thought or recall list item from which MediaLab is to supply the responses in combination with the *q* parameter. For example, say subjects listed their thoughts earlier in a variable you named 'tlist.' To have subjects later rate these thoughts using a thought rating item (for example how positive or negative each thought was), you would specify the parameter *(qtlist)* in the *Parameter(s)* field of your though rating item. Note that for thought ratings, you must also specify the number of scale points parameter 61.

ttl signal (optional)

To communicate with external computers or hardware you can send out a signal from MediaLab at the onset of any item. The signal can be a byte value (any integer from 1-255). You can send the signal to any port address you like. Make sure you use the decimal value of the port—not the hex value. For example, the port address of the parallel port on most systems is 888 (or 378 in Hex). Finally, you can send the signal for any duration you like before it resets to 0. Specify these three variables in the parameters field like so:

<value:location:duration>

For example, say you present an image item for 5 seconds and at its onset you want to send a signal of 255 to port 888 for 10ms. To present it for 5 seconds, you would use parameter value of *(d5)*. To add the TTL signal at its onset, you would add:

(d5,<255:888:10>)

Note that there is a TTL I/O test utility in the *Utilities* folder. You can use *IOTest.exe* to test whether your external hardware can send and/or receive TTL signals from any given

port. See also *Serial Data* 107

width, height (optional)

You can specify the width and height of the essay and fill-in-the-blank text boxes using the width and height parameters to allow for additional room for question wording or for background images/html, e.g., (h150,w400). To determine values appropriate for your display, select "Showing Location Points" under the main MediaLab Help menu.

You can also use the height and width parameters to specify the display size of images. Note that if you size an image using the height and width parameters you need to also specify the top and left parameters 60 otherwise it will appear in the top left corner by default.

By default, videos appear in their original size. However, you can set the exact size of the video using the width parameter. The height will automatically be set in proportion to the width you specify. For example, if you specify (w200) for a video that is normally 320x240 it will appear as 200x150.

You can express location and size parameters for ANY item type as a percentage of the screen. E.g., *(w.3,h.25)* would be a rectangle 30 percent of the screen's width and 25 percent of its height. To use percentages, simply use a value less than 1 and greater than 0.

You can also use the following shortcuts rather than specifying an exact width (e.g., w-1, w2, etc.)

- none or w0 = Default Size
- w-1 = Full Screen
- w-2 = Half Screen
- w-3 = One Fourth Screen
- w-4 = One Sixteenth Screen
- w1 = Double Size
- w2 = Half Size
- w10 and higher = specified width & proportional height

Examples

Scale Response. To set the number of response options for a scale response to 5: (p5)

Image. To set the top left corner of an image to the center of a 640x480 resolution screen: (l320,t240) or (l.5,t.5)

Instructions. To require that password be entered before the subject can continue (e.g., at the end of the experiment, between phases of the experiment), specify a password preceded by an x, e.g., (xmonkey).

Movie. To play a video file at full screen: (w-1)

Thought listing. To have subjects list up to 8 thoughts, or until 120 seconds have passed, which ever comes first: (n8,d120).

Background. For backgrounds, backsounds and backvideos, the parameters are always entered in parentheses after the filename and are always optional. E.g., For a background image to be displayed with the top left corner located at the top left corner of the MediaLab window, specify the background file as: myimage.bmp (t1,l1)

5.14 Position

QuickInfo

Identifies the order in which items are presented in the questionnaire.

Values

Any number. Decimal values are allowed.

Purpose

This field determines the order of the items in the questionnaire file. MediaLab will automatically sort the items in the questionnaire by these values when you close the file. You can also manually sort the items at any time using the *Sort* button.

Hints

To add a new item between two existing items (e.g., 2 and 3), assign it a position value between the values of the two existing items (e.g., 2.5) and then click the *Sort* button. After sorting, you can automatically renumber all of your items as sequential integers by clicking the *Renumber* button.

5.15 Randomize Within Groups

QuickInfo

MediaLab will randomly order and present all items sharing the same RWG value.

Value

Any positive integer value

Purpose

You may wish to randomly order and present certain items in your questionnaire file. To do this, assign the same RWG value to all the items you want to have randomized with each other. For example, if you have five items you want randomized, assign them each a *1* in the RWG field of each item. The five items will be randomly presented in the five positions they occupy in the questionnaire. All other files will be presented in their assigned positions as long as their RWG value is either *0* or *none*. You can independently randomize another group of items by assigning them all a *2*, another group with *3*s and so forth.

Example

In your questionnaire, you want to present 1) an instruction screen, 2) five scale response items randomly ordered, 3) three fill-in-the-blank items randomly ordered, and 4) a final Word document. Assign an RWG value of *0* or *none* to the instruction screen and

Word document to keep them where they are, a *1* to the five scale response items, and *2* to the three fill-in-the-blank items. That's it. To randomize the order that groups of items are presented (e.g., whether the scale items or the fill-in-the-blank items are presented first),see Randomize Between Groups 65 .

Note

For ease of analysis, variables will be written to the data file in the order the occur in the questionnaire, not in the randomized order in which they are presented.

5.16 Randomize Between Groups

QuickInfo

All items sharing the same RBG value define a group to be kept together. If multiple groups are defined, then MediaLab will present the groups of items in a random order.

Value

Any positive integer value

Purpose

You may wish to randomly order and present entire groups of items in your questionnaire (e.g., randomizing whether demographic or personality questions come first in a questionnaire). To do this, assign the same RBG value to all of the items you want to keep together. For example, if you have five sets of ten items, and you want to randomize the order in which these 5 groups of items are presented, then assign each group a different number in the *RBG* field and make sure all the items within each group share the same number for that group. The items within each group will all stay together, but the groups themselves will be randomly ordered. Items assigned a *0* or *none* will not move. Note that all items within a group must be sequential in the questionnaire.

Example

In your questionnaire, you want to present 1) an instruction screen, 2) five scale response items, 3) three fill-in-the-blank items, and 4) a final Word document. Whether the scale responses come first or the fill-in-the-blanks come first is to be random. Assign an RBG value of *0* or *none* to the instruction screen and Word document to keep them where they are, a *1* to each of the five scale response items, and a *2* to each of the five fill-in-the-blank items. When MediaLab runs that questionnaire file, the instruction screen will always come first, followed randomly by either all 5 of the scale response items or the three fill-in-the-blank items, and always end with the Word document item. To randomize the order of the items within each group, see Randomize Within Groups 64.

Note

For ease of analysis, variables will be written to the data file in the order the occur in the questionnaire, not in the randomized order in which they are presented.

5.17 Skip To

QuickInfo

The *Skip to* field allows the Editor to create simple or advanced skip patterns in their questionnaire.

Details

Conditional Skipping based on Scale Responses

After a scale response, MediaLab checks it for it corresponding *Skip To* value. If the value is *none*, it proceeds to the next question. If it has any other value, it assumes that value is a variable name and skips each subsequent question until it finds the one with the variable name that matches the *Skip to* value it's looking for. Skipped items will be assigned missing values in the data set.

Unconditional Skipping

In some cases you may want to skip over items unconditionally. Unlike conditional skipping, you can do this with any item type. For unconditional skipping from a scale response simply place the same variable name in each *Skip To* field of all the response choices. For all other item types, simply enter the variable name you wish to skip to in the single *Skip To* field.

Unconditional skipping is especially useful for rejoining subjects who have been conditionally skipped to various points in the questionnaire so that they end up at a common place. When they are done their special sections, they can be joined up to the same question by putting an unconditional skip item at the end of each of their custom sections. In each case they are all skipped to the same item and continue on with the same set of items.

Advanced Hints

Combining the skip function with *RWG (Randomize-Within Groups)* 64 you can achieve extra randomization functionality. For example, to randomly present 5 of 10 items give all 10 items the same *RWG* value (e.g., *1*). This will randomize their order. Then, embed a blank instruction item with a delay of 1 second after the fifth item and give it a skip value to go the item following the tenth randomized item. This will result in the five of the ten items being presented (which were randomly ordered) after which MediaLab will automatically skip the remaining 5 unpresented items and proceed with the rest of the questionnaire.

Skipping a Fixed Number of Items

As an alternative to explicitly naming the variable you want to skip to, you can simply enter *+1*, *+2*, *+3*, *+n*, etc. MediaLab will automatically skip that many items without needing to know the name of a *Skip to* variable--it will simply resume wherever it ends up. This is especially nice when the item being skipped to is randomly determined and so you do not actually know the variable name ahead of time.

Skipping on Multiple Response Items

You can include *Skip to* values with multiple response items but they work if and only if the subject chooses a single response. To be super safe, you can combine this with the *(r1-1)* parameter in order to disallow multiple options from being selected. For example, using *(p5,r1-1)* in the *Parameter(s)* field for a 5-item multiple response item will ensure

that only one of the five response options must be chosen and, consequently, guarantees your skips to work.

5.18 Text Labels

QuickInfo

Text labels provide the response alternatives (e.g., strongly agree, female, >10,000, and so forth, very positive) for scale response and thought rating items. Simply type whatever label you want in the *Text Label* field of a response option. If you specify *none*, for a given text label, then that label will appear blank. This can be used, for example, to create a scale anchored only at the ends.

Hints

You can also now include <specify> in the text label of any response option. This will not be displayed to the subject but will cause MediaLab to prompt subjects to enter a text response. This is handy in cases where you want to provide an alternative such as "other." For example, if you wanted a response option to be labeled as "other" and you wanted subjects to specify what "other" means, then enter other <specify> in the *Text Label* field for that response choice. The actual text response will be written to the data file.

Also note that copying and pasting a questionnaire item using the <specify> options, all the copied items will also contain the <specify> option.

Advanced Hints

If you want to specify a specific location for a button you can add top and left parameters 60 to the T*ext Label* field for that response option. For example, if you want to relocate a button with a text label of "skip this question" then you could type "Skip this question (t.7,l.7)" in the *Text Label* field for that response option. That would make the button appear 70% of the way across and 70% of the way down the screen.

If you want to adjust the size of the button you can add height and width parameters 63 to the T*ext Label* field for that response option. In the example above, if you wanted to resize the button as well as relocating it, you could type "Skip this question (t.7,l.7)" in the *Text Label* field for that response option. This would produce the same result as above except the label would be 20% of the screen width.

Finally, if you want to place the text label within the button itself, then simply put the text label in the *Label* field the parameters in the T*ext Label* field for that response option, like this:

	Label	Text Label	Skip to:
1	skip th	(t.7,l.7,w.2)	none

The result will be:

Inserting Variables in Response Options

You can also insert prior responses 76 from fill-in-the-blank items, scale responses, multiple responses, and any response.xls variables you have calculated 101 into the response options of any subsequent questionnaire item that allows response options. Simply type the item name 56 with a set of <>, like this: <*item_name*>. For example, if you have items named "friend1" and "friend2" you could have whatever was recorded for those items appear in the response options for a later item like this:

1. I would pick <friend1>

2. I would pick <friend2>

Whatever value was recorded for the "friend1" item will appear in place of <friend1>, and whatever value was recorded for the "friend2" item will appear in place of <friend2>. Remember that to insert a variable into an item, that variable must first have a response recorded in the data file.

Using HTML for Labels

If you want to create custom formatted or graphic labels for your scale response options, you can enter an HTML file or URL in place of the standard text label. This can be used to present formatted text or images beside each button.

Here is an example. Paste this code into a text file and save it as "label1.htm" in your experiment folder. You can preview the result by opening the file in Internet Explorer.

```
<body style="font: 12pt arial; color: navy; background: #ccccff; margin: 0px">
<i>Strongly</i> Agree
</body>
```

Enter "label1.htm" for your button label and this is how it will appear:

See also

Questoinnaire (.que) Files > Labels 55, especially for information regarding valid input options and recoding response data

5.19 Write to Data File

QuickInfo

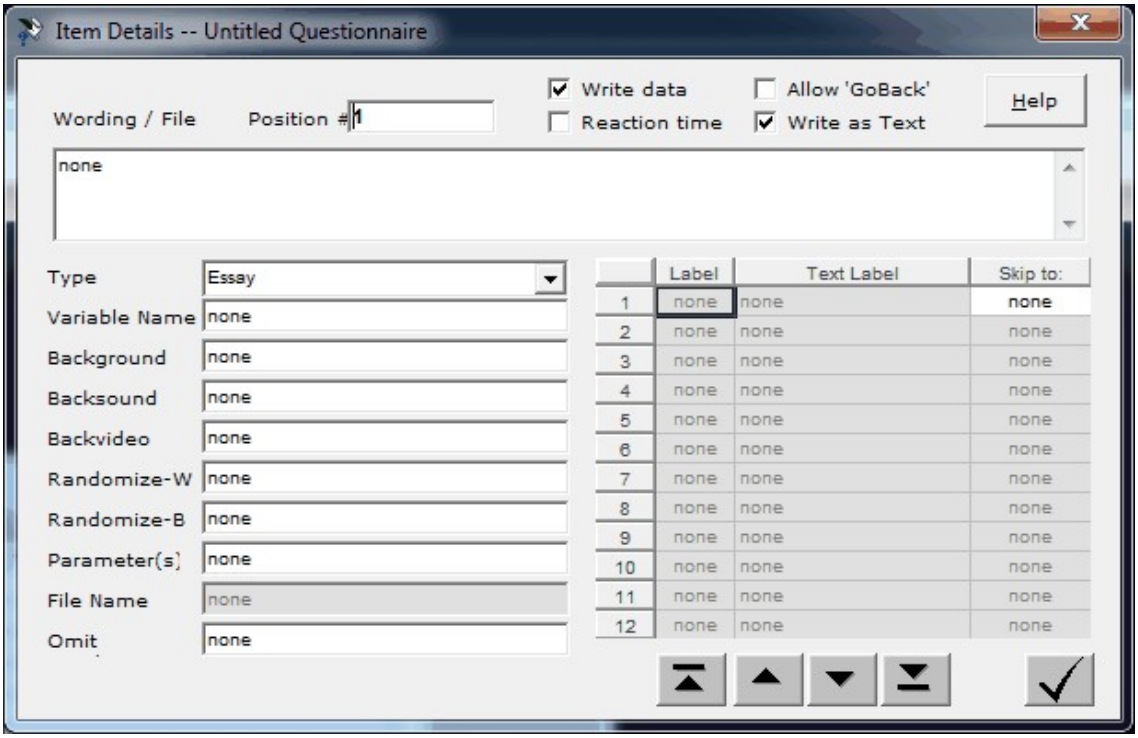
By checking this field, data will be written to the data file. When the value is unchecked, MediaLab will not write a value for this variable to the data file and will not include the variable name in the data input file. See Data 70 for more details.

For item types like Instruction items that do not gather input data, the response "ok" is written to the data file to confirm that the item was presented (i.e., as opposed to being skipped). If this you do not to write data from these items for whatever reason (e.g., practice trials), simply uncheck the *Write to Data* File field for that item.

5.20 Write Data as Text

QuickInfo

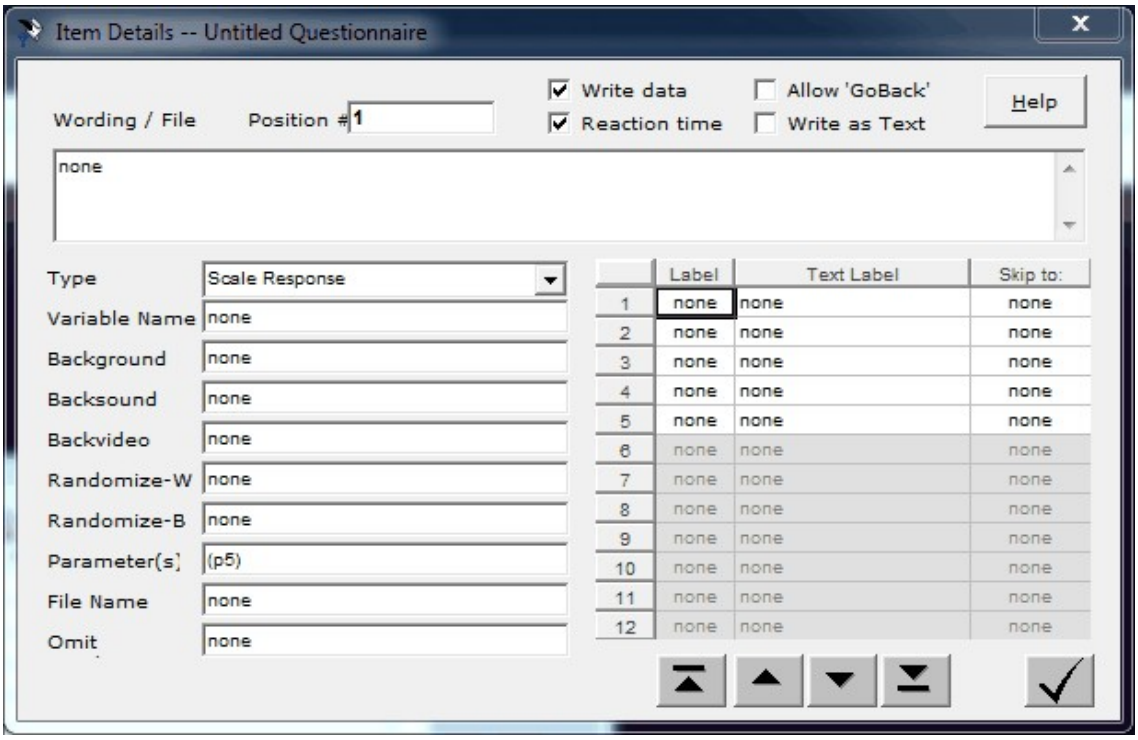
When the *Write as Text* field is checked in the Details menu (see below), MediaLab will write the data to the data file with quotes surrounding it. It will also indicate in the SPSS input file that the variable format is text; otherwise, value formats are recorded as numeric. Missing values for items with the *Write as Text* field checked will also be treated as text format; otherwise missing data values will be recorded as numeric format. Typically, this option is checked for item types like essay 44, recall list 49, and thought listing 51 whose data are primarily text, but it can be used with other item types as well. Also see the Data 70 section of the manual for more details.



5.21 Write Reaction Time

QuickInfo

When the *Reaction Time* field is checked in the Details menu (see below), the response time for the item will be written to the main data file. In the data input file which MediaLab creates, a "t" will be appended to the beginning of that item name 56 in order to create a new variable name for the response time. You can leave this field unchecked if you have no interest in assessing the response time for a given item. Also see the Data 70 section of the manual for more details.



Viewing Data

By default, data files are written to a *Data* subfolder located in the same folder as your experiment (*.exp*) file. For example, if your experiment were *c:\experiments\test\test.exp* then your data would be saved to *c:\experiments\test\data**. If this data subfolder does not exist when you run a session, MediaLab will create it. The main data file will be given the same name as the experiment file, but with different file extensions to indicate the different formats (e.g., *.txt* for the SPSS text-formatted data, *.sav* for the native SPSS data file and *.csv* for the comma-delimited text files). Data from essay items 44 and on-line rating items 48 will also be located in the *Data* subfolder.

You can view these files by browsing through "My Computer" on your desktop to the appropriate data folder. You can also select *View Data* from the *Data* menu in MediaLab--this will allow you to view any data file (.txt, .csv, .sav) you select. Note that MediaLab will use the default program your computer uses to view these file types.

Alternative Data Structures

How does MediaLab write the data file given that different subjects can get different dependent measures? Good question! This was a real challenge, because we wanted to be able to produce a *single* data file that could be a constant for all subjects no matter what condition they were in. As you may know, data are collected during the administration of questionnaire files (in which anything can be embedded). Subjects can then receive different questionnaires in different conditions, or the same questionnaires but in different orders. Before starting an experimental session, MediaLab goes through the *whole experiment file* and looks for all the questionnaires you have in *all* your conditions. When MediaLab writes the data file, it writes data for ALL the questionnaires in the experiment, whether or not the subject received them. If a subject did not receive certain questions or questionnaires then missing values are written. When the subject has finished, MediaLab writes TWO sets of data files that contain the same information but are structured differently depending on your needs. These two sets are called

ByQuestionnaire and *ByVariablename* and can be found in the Data subfolder of your experiment. Here is the difference between the folders:

ByQuestionnaire

The ByQuestionnaire data folder organizes your data by questionnaire (hence the name). Before writing the data, MediaLab alphabetically orders all of the questionnaires in the experiment. MediaLab then writes all of the data for each variable in the first questionnaire, the second questionnaire and so forth. If a subject did not receive a particular questionnaire in their condition, then missing values are written. Thus, the final data file lists variables sorted first by the alphaabetical order of the names of the questionnaire files used and then by the order of the items as they were programmed into the questionnaire file. This results in a constant data format no matter which questionnaires a subject received and the order in which they were administered.

ByVariablename

In contrast, the ByVariablename data folder organizes your data by variable names. Before writing the data, MediaLab goes through all of the questionnaires in your experiment and alphabetically orders all of the variable names. It then proceeds to write the data for all of the variables in this order—regardless of the questionnaire in which they occurred. Remember that the variables will be in alphabetical order so *q10* would end up following *q9;* If you are using a naming system like that, you might want to use *q09* instead. Because the ByVariablename data files don't care about which questionnaire asks the questions, the data for items with identical variable names will all be written to the same column even if they are asked in different questionnaires. Thus, the final data file lists all variables in alphabetical order of their item names, regardless of their questionnaire file or order of presentation during an experiment session. This results in an extremely easy way to analyze data from most experiments. It is also especially useful when you want to include the same variable in different questionnaires for different experimental conditions.

CSV vs TXT vs SAV Files

You will notice that MediaLab writes a *.txt* file and a *.csv* file to each folder. The *.txt* file is intended to be read into SPSS which has no practical limit on cases or variables. The accompanying *.sps* syntax file that is generated will read this *.txt* file. The *.csv* file is intended to be read in by pretty much any spread sheet application such as Excel. Many people prefer to use Excel method because it's simpler, but note that Excel is limited to reading in the first 255 variables from your *.csv* file *unless* you are using Excel 2007 or later.

Notice that there is also an *.sav* file located in the *ByVariablename* folder. This is a native SPSS data file that can be opened directly in the Data Editor Window of SPSS. You need to be careful with the *.sav* files because they are very sensitive to changes in the structure of your experiment. If you make changes to your study, it's usually best to delete (or move) the old *.sav* file and allow MediaLab to generate a fresh one for the revised experiment. Note that versions of SPSS prior to version 12 only allow variable names of up to 8 characters; more recent versions allow for up to up to 64 bytes in length. This typically means a maximum of 64 characters in single-byte languages, such as English, French, and German, and a maximum of 32 characters in double-byte languages, such as Japanese, Chinese, and Korean. If you have any names longer than the limit allowed with your version of SPSS (e.g., with suffixes added by MediaLab) then they will be renamed VAR0001, VAR0002, etc. In order to find out which variables had to be renamed, you can double click on them in SPSS. MediaLab saves the original variable name as the variable label.

Advanced Hint

Some people strongly prefer using the *.csv* data format but get stuck when they have more than 255 variables because they do not have Excel v2007 or later. If you try opening your *.csv* file in Excel and get the "file not loaded completely" message, you can deal with this by importing the file directly into SPSS. To do so, first rename the file from *.csv* to *.txt.* Then, in SPSS, you can select

```
File > New > Data

File > Open

Files of type > Tab-delimited (*.dat,*.txt)

File name  <select your file>

Predefined format? No

Delimited? Yes

Variable names at top? Yes

First case = Line 2

Each line represents a case? Yes

All of the cases? Yes

Delimiter? Comma (uncheck all others)

Finish
```

If you have any trouble reading in the data, you can look at the data directly in any text editor. If the file is especially large and difficult to read in its raw form, try a powerful free text editor like PSPad from www.pspad.com.

See also

Data FAQ 72

6.1 Data FAQ

- Merging Data 73
- Using SPSS to read your data 73
- Using Excel or other spreadsheet to read your data 73
- Getting Data into SAS 73
- Viewing Data 74
- Determining presentation order of random items and files 74

- Determining when events happened in real time 74
- Location of essay data 74
- Changing the values written to the data file 74
- Changing the value used for missing data 75
- The Clear Data button and how to disable it 75
- How to create a vertical format data file 75
- How to save data from a custom item made with HTML 76
- Using Long variable names 76
- Inserting variables into question wording and response options 76

Is there an easy way to merge multiple MediaLab data files?

Each subject's data are appended to the same constant data file for ease of analysis, so you won't have to merge any files that are collected on the same computer. You will however have to merge files collected on different computers. MediaLab offers a utility called FileMerge located in the *Utilities* folder to do this rather painlessly. You can also start FileMerge by selecting *Data > Merge Data Files* from the main MediaLab menu. After opening the filemegre application and a separate window on your computer containing the files you wish to merge, just drag the files you want to merge into the FileMerge window and select "Merge Files" from the File menu of the filemerge application. If you like, you can also save the list of files to merge them again later. A quick tutorial on FileMerge is available by clicking on the FileMerge Help menu.

How do I get SPSS to read the .txt data?

The easiest way to access your data in SPSS is to open the .sav data file located in the *byVariablename* data subfolder. However, if for any reason you want to import the raw text data files using SPSS syntax, here's how. If your experiment is called exper1, then the text file "exper1.txt" will have all your data in format that SPSS can read. The text file "exper1.sps" will be the input syntax file for SPSS to read. Just open "exper1.sps" in SPSS--make sure that you don't try to open it in the Data Editor window--and it will read the exper1.txt" file. It will automatically assign variable formats and variable names. You can start analyzing immediately. At the top of the .sps input file are a number of things to check for to make sure you don't get any errors (e.g., variable names that are too long). Once the data has been read into SPSS, you can save it as a .sav file, analyze it immediately or export it into almost any other database format.

How do I get Excel to read the data?

If your experiment file was named "exper1" then also written to the *Data* folder will be a file called "exper1.csv". This file will have all your data in a format that Excel can read. Just open this file with Excel, and it will automatically read in the data, variable formats, and variable names. You can then save the file as an .xls file readable by almost any stats package. Be careful though, Excel can only read 255 variables. If you have more variables than that (including response times), then you may want to have SPSS read in your file.

How do I get SAS and other data analysis packages to read the data?

Most stats package will have the capability to read the .csv file (this is the comma-delimited file) in the data folder of your experiment directory and transform it into format

compatible with itself. If not, have Excel read the .csv file and then have your stats package read the Excel file. Some will require you to save the Excel file in an earlier version (e.g., as an Excel 2003 worksheet).

How do I determine the order in which random items were presented?

There is routine that keeps track of the actual presentation order of everything (e.g., of files and questionnaire items that are presented randomly). After completely running an experiment file in Medialab, you can find a log file called *question_order.csv* in the *ByQuestionnaire* 71 data folder. The first row is the order of files (i.e., from the *.exp* file) and the remaining rows are the question orders for each questionnaire in the order they actually occurred, with a time stamp written to verify the time at which each questionnaire was started.

How can I tell exactly when an item was presented—i.e., in real time?

By entering *@ in the Parameter(s) field* 59 for any item in a questionnaire, you can have MediaLab write a timestamp at the onset of the item. When you do this, a timestamp will be appended to a file called *timestamps.csv* in the data folder. It will include the subject id, the variable name, the time at which the item *started* (e.g., 12:23:22), the milliseconds elapsed from the beginning of the experiment and the milliseconds passed since the last timestamp.

Where is my essay data?

Because large quantities of text do not play well with most data files, the responses of subjects to essay items 44 will be written to their own separate data files located in the main *Data* folder. The file will be named after the item name you use for the essay. For example, if you have an essay item named *ess1* then look for your essay data in a file called "ess1.txt" in the main data folder. The responses of all subject to this question will be appended there along with their subject and condition IDs.

Can I change the values written to the data file?

On scale response items 50 and thought rating items 51, the default value written to the data file is always the ordinal value of the response option. For example, if the first option is chosen then a *1* is written to the data file—even if the button label read *A*, *-5* or *0*. For multiple response items, the default value of a chosen response is always *1* or *0* if that response option is not chosen. For all of these item types you can specify an alternate value for any given response option. This simply saves you from having to recode the data after collecting it. To specify an alternate value, specify it in squiggly brackets after the text label like so:

	Label	Text Label	Skip to:
1	1	strongly disagree {-2}	none
2	2	disagree {-1}	none
3	3	neutral {0}	none
4	4	agree {1}	none
5	5	strongly agree {2}	none

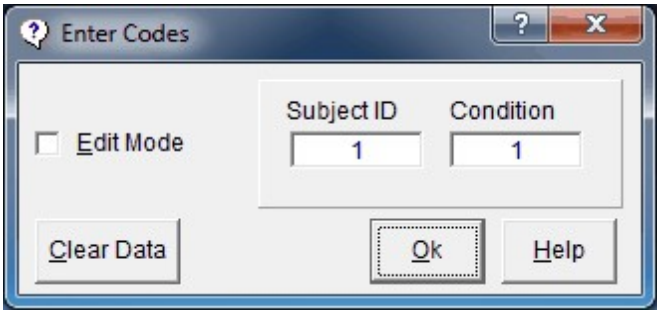
Note that the value can be text or numeric although numeric is recommended. If you use text, make sure the *Write as Text* 69 option is checked for the item.

Can I change the value used for missing data?

By default, 99 is used as the value for missing data. Because most item types are either open ended or limited to a numeric value of 12, 99 is usually safe. However, if you suspect 99 might be a dangerous value for your particular study you can choose an alternate value. Simply create a text (.txt) file in your experiment folder and name it "missing.txt" and on the first line of that file, enter the alternate value. It must be numeric.

What is the Clear Data button for and can I disable it?

When you make changes to your experiment, you often change the structure of your data. This can cause confusion in your data because one subject's data are always appended to the last. Consequently, it is always a good idea to clear your data folder after making changes to your study. If the existing data are important, then back them up and merge them later [73] with the new data. If they are not important (e.g., they are just test files) then you can click the *Clear Data* button when entering the subject and condition IDs after selecting a file to run [9]:



This will delete everything in your data folder resulting in a clean start. Note that some people get very nervous about having this button so accessible. If that's you, there is a way to disable it—on the Preferences menu, simply click *Disable Clear Data*. Note that MediaLab will attempt to send cleared data to the Windows *Recycle Bin* if the experiment is stored locally (i.e., as opposed to on a network drive).

How do I create a vertical format data file?

Occasionally, you may want to access your MediaLab data in a vertical format--i.e., one variable per row--rather than the usual horizontal format--i.e., all data for each subject located in a single row. Located in the *Utilities* folder is a file called *vertical.mdb*. This file is a blank Access database. This file can be copied into the *ByVariablename* [71] subfolder of any experiment you like. If MediaLab finds this file after a session is complete, it will output all of the data into three columns (subject, varname, value) of the "vertical" table. MediaLab will always append data to whatever data already exist in the file and this function will be in addition to all of the usual data file formats. To maximize compatibility, this file is in Access 97 format--consequently it is readable by all versions of Access. A simple test of this function would be to copy the *vertical.mdb* file into the *ByVariablename* [71] subfolder in your experiment folder and run that sample--you should find the results stored in that file after you run the session. This is also a handy way to programmatically access data immediately after a session for use in other programs.

How to save data from a custom item made with HTML

See Custom Items 88.

Can I use Long Variable Names?

The limit in MediaLab for variable names is 60 characters. Keep in mind though that some statistical packages and spreadsheet applications may still have difficulty working with longer variable names. Be sure to view and analyze some practice data before going too crazy with this feature!

Inserting variables into question wording and response options

You can insert variables from the data files into the question wording and response options by using <*variablename*>. For example, if you have variables in your data file called "friend1" and "friend2" you could have response options for any subsequent items in your questionnaire files like this:

1. I would pick <friend1>

2. I would pick <friend2>

This will work to insert prior responses from fill-in-the-blank items, scale responses, multiple responses, and any response.xls variables you have calculated 101.

See Also

Data 70

Tutorial and Additional Samples

A good way to get the feeling of how MediaLab works is to run a couple sample experiments and then look to see how they were put together. If you are new to MediaLab you can get the gist of things by running through the quick on-line tutorial. From the main menu in MediaLab, select *Help > On-Line Tutorial*. The text from the six steps of the tutorial are also contained in this section (see Sample 1 77) in case you would rather work from a hard copy (if you are reading this online, you can always right click any topic and select *Print*).

After that, you can take a look at any of the additional sample experiments that illustrate the basic features of MediaLab. They are contained in the *Samples* folder of the MediaLab program folder. With the exception of the first basic sample, all the samples are zipped and will need to be *unzipped* in order to use them. If you are using Windows XP or later, you can do this by right clicking on the zip file and selecting *Extract All.*

Other Sources for Samples

www.empirisoft.com/support - see MediaLab: Shared Experiments

> *This area of our website forums is provided for you to post your MediaLab experiments, questionnaires and other MediaLab related files for others to play with. Feel free to post away! Also, feel free to comment on any posts if you happen to try them--sort of an informal peer review process. We may edit posts when we deem it necessary. Please zip your files and folders whenever possible! Note that you will need to register to download files, but registering is quick, easy and free.*

Continue to Samples 77

7.1 Samples

Sample 1 – Sample1.exp (text from *Tutorial* on the MediaLab *Help* menu)

"So I've installed MediaLab. Now what do I do?" Good question. This brief tutorial (about 15 minutes) will walk you through the basics of MediaLab. After you've run through this short tutorial you will likely be able to design and execute a short experiment of your own. This on-line tutorial can be run at any time by clicking *Help > On-line Tutorial* from the main menu. You might find it helpful to print out this tutorial so you will be unobstructed as you go through the steps of running and editing the sample experiment.

This purpose of this short tutorial is to give you the *gist* of how MediaLab works. After running through it, you may wish to check out the additional sample experiments and questionnaires located in the *Samples* folder of the MediaLab program folder (usually *C:\MediaLab\Samples*). For more detailed help, select *Help > MediaLab Help* from the main menu of MediaLab or press *F1* at any time. You will also find a printable *PDF* manual located in the *Help* folder (usually *C:\MediaLab\Help)* called *MediaLabManual.pdf.*

Step 1: Run the Sample Experiment

The first sample experiment is as simple as MediaLab gets—a basic questionnaire, in this case measuring self-esteem. To run Sample1.exp:

1. First, if MediaLab is not running full-screen, then make it so by clicking the maximize button in the top-right corner of the main MediaLab window (it's the one in the middle).
2. Go to the *Run* 9 command in MediaLab and choose *Select and Run Experiment*.
3. Find the *Samples* folder located in the *C:\MediaLab* directory.
4. Double-click on the *Sample1* folder, and then double-click on *Sample1.exp*. Note that experiment files 20 are indicated by a green "building block" icon and that questionnaire files 34 are indicated by a "paper and quill" icon.
5. Enter a *Subject* ID (e.g., *5*), and press *Enter.*
6. Enter a *1* for the experimental *Condition*, and press *Enter.*
7. Follow the instructions as if you were a participant.

Step 2: See How it Was Done

From the main menu in MediaLab select *Experiment Editor* 11.

When the Experiment Editor opens, select *Open* from the *File* Menu and open *Sample1.exp*.

You'll see that there are 2 conditions 25. Everybody in condition *1* gets only one file—*selfesteem.que*.

However, in condition *2* you see that these subjects will also get a *PowerPoint Show (.pps)* file before they get the self-esteem questionnaire. In MediaLab experiments, you can define as many conditions as you like, with each having as many files as you like. These files can be the same ones in different orders, or they can be different files altogether.

That's all there is to this experiment.

Close the experiment file by clicking *OK*. Then close the Experiment Editor. This will return you to the main MediaLab window. Now run the experiment, enter a subject ID, but ask for Condition *2* this time. After that...

Step 3: See How the Questionnaire Was Made

From the main menu bar in MediaLab select *Experiment Editor*.

When the Experiment Editor opens, select *Open* from the *File* Menu and open *selfesteem.que*.

Here you'll see the items contained in the questionnaire file. Here you can see the basic information about each item—the ordinal position 64, the item name 56 and the question wording 53 or filename 41. Double-click on any position number to see the item details. Note how *Scale Responses* require the p parameter 61 telling MediaLab how many *scale points* the question requires. Try using the *context sensitive help* here. Place the cursor on any field (e.g., the *position* field) and pressing *F1*. Close the Details window 36 by clicking on the big checkmark in the lower right corner, and then click *OK* to close the questionnaire and then close the Experiment Editor.

Step 4: Look at the Data

Your data 70 will be located in a data folder contained in the same directory as your experiment. From the *Data* menu in MediaLab, select *Launch Explorer*. Find the *Sample1* folder and open the *Data* folder located inside. In here, you will see two folders. Usually, you will use the byvariablename data folder 71. Take a look inside and you'll see a number of files. MediaLab has generated all of these files for you automatically. The first one to look at is *sample1.csv*. You can open this with Excel. Try it. (note that all the variables that start with t are response times 69).

If you've got SPSS installed, you can open *sample1.sav* 73. Or you can open the .sps syntax file, highlight everything in the file and run it. This input file tells SPSS to read the data file *sample1.txt* (also located in your data folder), and defines all of the variables. After running this input file in SPSS, you can analyze it right away. Try doing some frequencies or means.

The other file in the main data folder is called *comment.txt*. You can open this in any text editor. This file contains the open-ended responses 74 from the essay item called *comment* in the self-esteem questionnaire. MediaLab writes the data from *essay* questions to their own file since it would get out of hand to write these to a standard data file.

Step 5: Modify and Run the Experiment:

Now let's try to modify the experiment you just ran. With Windows Explorer still open, make a copy of the *selfesteem.que* questionnaire file and rename it *myquestionnaire.que*. Then, double-click on it to open it in the *Experiment Editor*.

This is where you can have some fun. Try changing the question wordings. Double click on any position number 64 to see the details for any item. Try this with a Scale Response item 50. See where it says *(p6)* in the *parameters* field? Try changing this to *(p9)*. See how you now get 9 response options for the questions? You can add the additional options in the *Label* 55 and *Text Label* 67 fields.

Close the *Details* window by clicking on the big checkmark in the lower right. Try adding an item of your own. In the first open position, enter a new position number of *4.5*. Enter an item name 56 such as *MyVar*. Double-click on the *ItemType* field and choose *Fill-in-the-blank* 44 from the pull-down menu. Finally, enter your question wording 53 in the *Question Wording/File Name* field, e.g., "Please tell me what your favourite food is."

Now click *Sort* 37 and then *Renumber* 37. See how the item is now positioned between the 4th and 5th items? This happened because you gave the question a position value of

4.5.

Now click *Save* 38 and *OK* to close the questionnaire.

Now we need to add your new questionnaire to the *experiment file* 20. From the *File* menu select *Open* and open the *Sample1.exp* experiment file.

Let's replace the selfesteem questionnaire in Condition 1 with your new questionnaire. Simply double-click on *selfesteem.que* in Condition *1*, Position *1*. When the dialog box opens, select you new questionnaire. See how it has replaced the old questionnaire?

Let's add another file to Condition 1 in addition to your questionnaire. In the first available row of your experiment file and enter a value of *1* in the *Condition* 25 field. Enter a value of *2* in the *Position* field. Double-click on the *File Name* 41 field and select an *image file* on your hard drive (and jpg, bmp or gif file). To see the types of files you can add here, click *Files of Type* when the dialog box opens. If you have Internet Explorer 4 or later installed, you could also enter a web address (e.g., www.empirisoft.com) here or a local *HTML* file. After selecting a file, click *Sort*, then *Save* and *OK*.

That's it. Close the Editor and return to the main MediaLab window. Choose *Select and Run Experiment* 9 and run Sample1, Condition 1 again. Note that if you have made changes that will affect your data files, you'll want to click the Clear Data button 75 to delete the old data files and start fresh (after backing them up if there are data you want to save).

That's the gist of MediaLab. Of course experiments can get much more elaborate than this but the basic idea is always the same. The experiment file defines the files that will be presented in each condition. These files can be pretty much anything including MediaLab Questionnaire files which allow you to ask questions. For more detailed help, be sure to check out the manual (*MediaLabManual.pdf*) contained in the MediaLab *Help* folder and to use the OnLine Help by pressing *F1* at any time. More samples are contained in the *Samples* folder.

Sample 2 - Scale Responses and an Introduction to Parameters

`C:\MediaLab\Samples\Sample2.zip` *(requires unzipping)*

This sample demonstrates what you can do with scale responses. It starts with the most basic application and ends with a relatively advanced example. You will need Microsoft's Internet Explorer, Version 4 or later installed to view the HTML files that are used in this sample. Click Continue or press the spacebar to begin. Press Ctrl+Alt +Right 10 to exit the sample at any time.

Standard Scale Item 50

The first scale response you'll see is the standard format which uses just the point (p) parameter 61 to indicate the number of scale points (see the Parameters field). This tells MediaLab how many scale points you want. In this case, it's 6. Click *Continue* or press the space bar to see the item. When the item appears, you can click *Go Back* 42 to see this screen again or enter a response to continue.

Inserting Prior Responses in Question Wording and Skipping

With scale items and fill-in-the-blank items, you can enter subject's responses in later question wordings 76. For example, what was the response you just gave to the last question? Whatever it was, it should show up in the next item. The question wording specifies <rse9>. That tells MediaLab to insert the subject's response to the item with variable name "rse9" into the question wording of this item.

Also notice that we're using "skip to" values here. The values in the *Skip To* field allow

you to skip a subject to any subsequent questionnaire item by entering the item name in the *Skip To* field. Below, you can see the Details for the next three questions: rse10, rse11 and rse12 (you may have to scroll down). To illustrate, say we want people who disagree with rse10 to answer question rse11 and we want those who agree to rse10 to answer question rse12. To do this we enter "rse12" in the *Skip To* field of the agree responses in rse10. If the subject agrees to rse10, they will be skipped to rse12, otherwise they will continue on to question rse11. Scroll down here to see the details of rse11. See how the subject is unconditionally skipped to rse13? This ensures that they will not get rse12 which was designed for those agreed to rse10. If that's confusing try going back and forth using the *Go Back* button trying the different combinations to see how it works. You'll see that depending on what you answer to rse10, you will get only rse11 OR rse12. Instead of a single item, you could also have entire blocks of items designed for a particular response.

Multiple Response Item

Sometimes you may want a subject to be able to check more than one response. That is when you can use the multiple response item 47 instead of the scale item. It works exactly like a scale response except that MediaLab will allow multiple buttons to be "checked". In this example, try clicking multiple answers. When you're done, click Continue or press the spac ebar. Multiple Response items use all the same parameters as Scale Items 58. Note that on this item we turn off the QuickStyle 84 we applied earlier using the ($off) parameter. You should see that MediaLab reverts to the default colours and fonts.

Inserting HTML in Place of Question Wording and/or Response Labels

Want more control over formatting? You can create an HTML file and display it in place of the usual question wording and even the text labels on the response buttons 67. In the experiment folder, there is a subfolder called "sp". This folder contains the HTML files and media. In the *Background* 39 field, see that we specify that we want MediaLab to display "sp/rse.htm" --an HTML file in the SP folder. When we do this we have to specify the area for it to cover using the top and left 60 as well as the width and height 63 parameters. We are also going to display html files in place of the response labels. To do this, we enter the name of the HTML files in place of the usual Text Labels. To allow for more space to accommodate the larger HTML labels, we have triple spaced the buttons using the spacing parameter 62 (s3). We have also specified that the scale buttons be moved over to the right and upwards using the Left and Top parameters. Finally, since the HTML files have a white background, we have applied a QuickStyle 84 file (white.mlq) here that instructs MediaLab to use a white background. This way the HTML files will blend in.

End of Scale Item Sample

That's it. This sample represent much of what you can do with Scale Items. Of course, everything you've seen here is optional aside from the use of the Points (p) parameter telling MediaLab how many scale points you need (e.g., as in the first question, "rse1"). Hopefully, if you need more options than that, you've seen here how to accomplish what you need. If you have any questions, feel free to contact us at support@empirisoft.com.

Adding Response Options & Modifying Button Labels

The next item is pretty much the same except that we've changed the Point parameter to 9, providing for 9 response options. We've also changed the button labels 55 to include negative values. Since there are no "negative" keys on the keyboard, you would need to label your keyboard appropriately (e.g., with 1 or F1 corresponding to -4, 2 or F2 as -3 and so on). Click *Continue*, or press the space bar to see the item.

Horizontal Response Scale

The next scale item is the same as the first except we now add the width parameter 63. This tells MediaLab we want a horizontal scale and specifies that it should be 500 units wide. See "Show Location Points" in the Help menu for the sizing scale that is used with your display. MediaLab tries its best to use a resolution-independent display. Click *Continue*, or press the space bar to see the item.

More Room for Question Wording

The next item again is the same as the first, except now we add the top parameter 60 which tells MediaLab that you want the buttons to start at a specific distance from the top. With this, you can move the buttons up or down as much as you like. Note that you can use the top and width parameters together to create a horizontal scale at a specified height. Click *Continue*, or press the space bar to see the item.

Getting Subjects' Attention

Sometimes after answering many questions, the items can start to "blend in" with one another. A couple things you can do to minimize this: You can add an onset parameter 61 which delays the display of the question wording. In this case we've added (o1000) which means there will be an approximate 1000ms delay before the question wording appears. We've also added a QuickStyle parameter 59. QuickStyle files 84 can be created in the MediaLab preferences menu. They contain font and color information. In this case, we apply a style file called "green.mlp" located in the experiment folder by entering the parameter value of ($green).

Creating More Room For Response Labels

In this item, we have some very long response labels. To create more space for them, we can double, triple or even quadruple the spacing of the buttons using the spacing parameter 62. In this case, we've entered (s2) to double space the buttons. Any value-decimals included-between 1 and 5 is acceptable.

Turning off the QuickStyle and Imposing a Time Limit

When this item occurs, we shut the previous QuickStyle off by entering a QuickStyle parameter of ($off). An alternative would be to apply a different QuickStyle file. Using the ($off) value returns the colours and fonts to the default settings in your preferences file. We also add the duration parameter 60 here to impose a time limit on the question. In this case, you will have up to 5 seconds to respond. Try letting the time pass and see what happens.

Adding Other Media

It's easy to add sound, images and video to your questionnaire items using the "Back" fields. In this case, we add an image in the *Background* 39 field and a sound file in the *Backsound* 40 field. In this case, we have entered parameters for the Background image immediately after the filename (this is always optional). This will locate the image with a left value of 300 and a top value of 100.

Adding a "Specify" Response

In some case you may want to offer an "Other" response. If you add <specify> to any response label, then MediaLab will prompt subjects to specify their answer if they choose this button. Try it. When the item appears, select the "Other" response and see what happens. If a subject specifies an answer like this, then the text response is written to the data file instead of the button they pressed.

Back to Samples 76

QuickInfo

The Preferences menu in the main MediaLab window provides a way to edit and save default and alternative program options.

Two types of files can be created and edited here. The first is the *MediaLab Preferences File* (*.mlp*) which contains font and colour combinations as well as a number of general program options. The second is the *MediaLab QuickStyle File* (*.mlq*) which contains ONLY font and colour information. Normally, you would have just one Preferences file (usually *default.mlp* which is loaded automatically) and many QuickStyle files. This is because you can only have one Preferences file loaded during an experiment, but you can apply multiple QuickStyle files throughout your experiment by requesting them in your questionnaires or experiment files. If you do not create or request any QuickStyle files during your experiment, then the default fonts and colors in your Preferences file will be used.

Preferences 83

Fonts and Colors 83

On-the-fly-editing of Word Documents 83

Hide MediaLab Rats 83

Disable System Key Sequences 83

Double Click to Close MediaLab 83

Use Alternate Data Folder 83

Use Unique Machine Code 84

Password to Exit Experiments 84

Where to Find Supporting Applications 84

Load Other Preferences File 84

Disable Clear Data 84

Mute MediaLab Sounds 84

Quick Styles 84

Character Set & Foreign Fonts 85

Changing Default Feedback and Message Text 86

AutoStart files to Automate Subject and Condition IDs 87

8.1 Preferences and Options

Fonts and Colors

You can change the fonts and colors MediaLab uses during experimental sessions. Select *Edit Preferences > Font & Colors*. You can set fonts and colors separately for windows that display text provided by you and for windows that allow subject input.

On-the-fly Editing of Word Documents

If this option is checked, then you can edit Word or WordPerfect documents as you run through the experiment simply by double clicking on them. You may want to disable this when you are running actual subjects.

Hide MediaLab Rats

For the politically correct and the phobic--you can choose to have MediaLab hide the rats from the main window with this option. For the phobic, note the opportunity here to start your systematic desensitization!

Disable System Key Sequences

Selecting this option will disable any key sequences 10 that would allow participants to explore your computer or otherwise interfere with your experimental session. The key sequences such as ALT+TAB, CTRL+ALT+DELETE, and CTRL-ESC will be re-enabled after the session is over. Caution should be used here such that you have fully tested your experiment before setting this option. If for any reason, the program locks up while these system keys are disabled, you would have to turn the computer off and restart it (e.g., since CTRL-ALT-DELETE would not be an option for closing the program). Note that this function may not work on Windows XP.

Double-Click to close MediaLab

A small convenience that allows you to double click on the blue area of the MediaLab window in order to close the program.

Use Alternate Data File

Click the button to its right to browse for a folder on your computer or network. MediaLab will write the data to this folder instead of the experiment folder (default). This is especially useful if you want to run your experiment from a CD-ROM.

Also, because you can specify any location for the data using the "alternate data folder," you can choose to gather all of your data on a central computer on your network. To specify another computer as the location for your data files you need to "map the drive" on the other computer as a drive letter on your computer running MediaLab (e.g., as h:\ or s:\ --you may want to ask your network administrator to help you with this if you are unfamiliar with mapping drives on other computers). You can then specify the alternate data folder by using this mapped drive, e.g., "h:\medialabdata."

When MediaLab sees this, it will create a folder within the alternate data folder with the same name as the experiment file, e.g., "h:\medialabdata\experiment1" and will write the data as it normally would to this folder. Please be aware that specifying a data folder that is not located on an actual machine in the network (i.e., Dropbox, cloud storage) would require an active internet connection anytime that data is being gathered. Otherwise, this could result in an error when trying to run your experiment because MediaLab will not be able to locate the folder to which it is attempting to write the data.

If you have multiple computers write data to this same alternate folder, it is

recommended that you use a "Unique Machine Code" (see below) for each computer. This 3-letter identifier of each computer will be appended to all of its data files. This serves to identify which data files came from which computer and also helps to prevent conflicts that may occur from multiple computers attempting to write to the same files at the same time (since each computer will write to its own unique files). These multiple files can be easily merged later using the FileMerge utility provided in MediaLab's data menu 73.

Unique Machine Code

Type any extension here that you want appended to data files produced on the current computer. This helps to identify the machine on which data was collected. Leave this field blank if you don't want a code appended to the data file names (default).

Password to End Option

You can set a default password that will be required at the end of every experiment. This saves you the trouble of creating a separate password item in your experiment. If you forget the password, the secret key sequence CTRL+RIGHT ARROW will skip you out of it. No password will be required if this field is left blank.

Where to Find Supporting Applications

For most users, these fields can be left blank because MediaLab will automatically know where these applications are. However, if you get an message saying that MediaLab can not find one of these applications then you can fill in these values and MediaLab will search there. You can click the "..." button to the right of each field to browse your computer.

Saving the File

Click Save to save your work or click Close to cancel the changes. Click SaveAs to save the current configuration to a new Preferences file. When clicking SaveAs, you can also then select "Save as Type" and save the colours and fonts to a new Quick Style file if you like (see below).

Load Other Preferences File

Allows you to load the preferences saved in another file. The default.mlp preferences file is always loaded when MediaLab starts. You can edit and save the default preferences under a new name and then load your new preferences file before you begin running your experiment. This can be useful if different members of your lab have different preferences for how MediaLab operates. Remember that if you're only changing colors and fonts, then it's much easier to just create a new QuickStyle file (see below).

Mute MediaLab Sounds

Upon certain events MediaLab will give audible feedback by way of beeps and clicks. To turn off these sounds check this option.

Disable Clear Data

Prevents the *Clear Data* 75 button from being pressed at the start of a session. This is a handy option if sessions are being started by users who are unfamiliar with the program.

QuickStyles: Create and Apply Multiple Style Files

MediaLab automatically loads the default preferences file. If you want load a different preferences file, you can select "Load Other Preferences File" from the Preferences menu.

You can create, edit and save as many Preference files as you like.

Quickstyle (.mlq) files are just like Preference files (.mlp) except that they contain ONLY font and color information. You can create many quickstyle files with various foreground and background color combinations as well as font options. The benefit of quickstyle files is that you can instruct MediaLab to apply a different quickstyle file at any time during the experiment or in the middle of a questionnaire. In the parameter field of any experiment or questionnaire item you can name a quick style file preceded by a $ which will cause MediaLab to switch to that style. You are no longer limited to using a single set of color and font options.

How QuickStyles work

To create a quickstyle (.mlq) file, select QuickStyles from the Preferences menu. You can then select a configuration of fonts and colors. Select *SaveAs* and save the quickstyle file with a name of your choosing (e.g., mystyle1.mlq). You can either save it in the MediaLab Styles folder so that all experiments can use it, or you can save it to your experiment folder to make it available only to your current experiment.

Then, in your questionnaire or experiment files, enter the name of the style file preceded by a $ in the Parameter(s) field 59 of any item, e.g., ($mystyle1). When MediaLab sees this it will read the style file and apply it immediately. MediaLab will first look in your experiment folder for the file (e.g., mystyle1.mlq). If it doesn't find it, it will look in the MediaLab Styles folder. If it finds it, it will immediately apply that style to your running experiment. You can have as many quickstyle files as you like and can apply them however frequently you like. The currently applied style will remain in effect until you request a different style file, or until ($off) is encountered in the parameter field of another item.

Aside from added flexibility of formatting, this allows you to store color and font information within your experiment folder so that it's always available. If for example, you send someone your experiment folder, the formatting will remain the same so long as the quick style files are included in the folder. This also means you no longer have to worry about loading the correct Preferences file before each experiment because your experiment takes care of this for you.

Note that you don't have to use quick style files at all. If you do not apply any quick style files in your experiment, MediaLab will simply use the colors and fonts specified in the Preferences file (e.g., default.mlp which is loaded automatically).

Foreign fonts

Improved foreign character support

If you need to display foreign characters in MediaLab there are now a few options that can help. In MediaLab, select *Preferences* and then *Edit Preferences* to choose the a compatible font for use when running your experiment (MediaLab supports most double-byte fonts such as *Arial*). Also under Preferences, you can select the *Character Set* option to select a script that works best with your language and font. Finally, also under *Character Set* is an option to *right-align* the text for question wording and open-ended responses. Finally, in the Experiment Editor, under *Options*, you can select an Editor Font and chose a font that works with the characters you need while editing your experiment. This step should be taken *after* setting the font and script preferences in the main MediaLab window.

Note that if you are designing experiments in a language other than English, you can also modify the messages that MediaLab displays so that they are in your preferred language (see below). We can also help you with a custom *Continue* and *Go Back* button for your language—let us know if you need this. In fact, there is already a series of *Continue* and *Go Back* graphics in various languages located in the *C:\MediaLab\Graphics* folder. If you

would like to use them, simply replace *C:\MediaLab\Graphics\continue.bmp* and *C:\MediaLab\Graphics\goback.bmp* with the ones located in the language subfolder of your choice (e.g., Chinese, German, Greek, Japanese, Spanish).

Modify message text

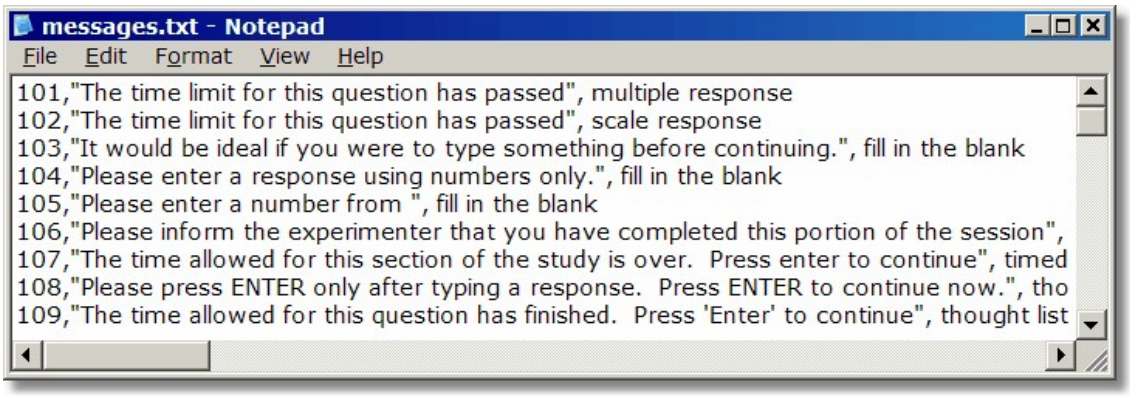
MediaLab presents various messages during an experiment. You can edit the content of these messages to suit your specific needs or even translate them into an alternate language. You will find two files in the "styles" folder called "messages.txt" and "substitute.txt" which you can edit in any text editor. Messages.txt contains all of the possible feedback that MediaLab presents to subjects during an experiment:

How to modify MediaLab messages

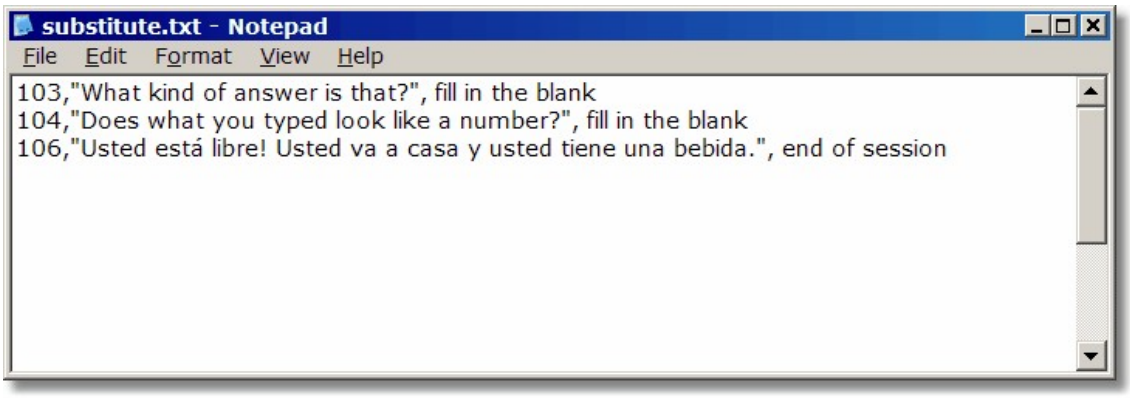
Do NOT modify *messages.txt*! That would be bad. Instead, create and modify a file called *substitute.txt*

If you would like to customize or translate messages simply copy the messages you want to change into *substitute.txt*. Make sure you copy the entire line including the ID number. Change only the text within the quotation marks. MediaLab will substitute messages based on their ID number. The substitute file you make may contain all of the messages in this file or just a subset that you want to change. You can put the *substitute.txt* file in the *C:\MediaLab\Styles* folder or in an experiment folder of your choice.

For example, you could copy messages 103, 104, and 106 from "messages.txt":



And copy them to "substitute.txt" where you can change them to anything you like:



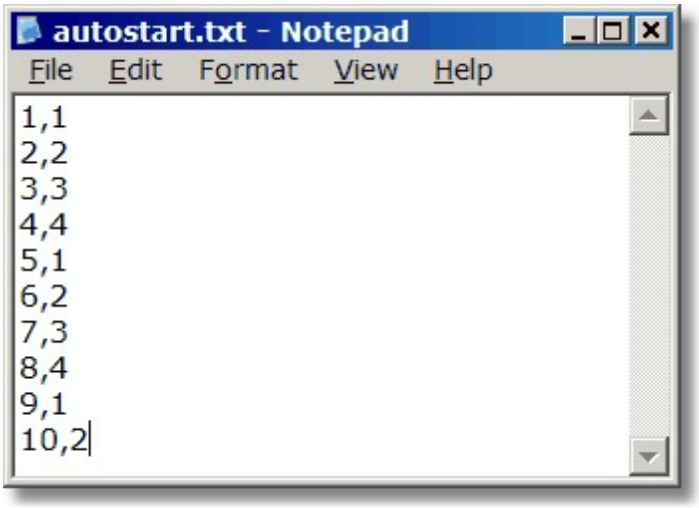
If you were to do that, then these messages (103,104,106) would be used instead of those contained in the *message.txt* file.

Here's how it works:

Every time MediaLab starts an experiment, it accesses the messages it will use from the main *messages.txt*. It will then look in the *Styles* folder for a file called *substitute.txt*. If it finds it, it will read your custom/translated messages and substitute them in the experiment. Note that if you place the *substitute.txt* file in the *Styles* folder, then all experiments will use the substitute messages. If you want to substitute messages in a particular experiment, then simply place the *substitute.txt* file in your experiment folder. Finally, you can place one *substitute.txt* file in the *Styles* folder (which will affect all experiments) and another in your experiment folder (which will affect only that experiment). If the same message IDs are contained in both locations, priority is given to those in the experiment folder.

Using an AutoStart File to Automate Subject and Condition IDs

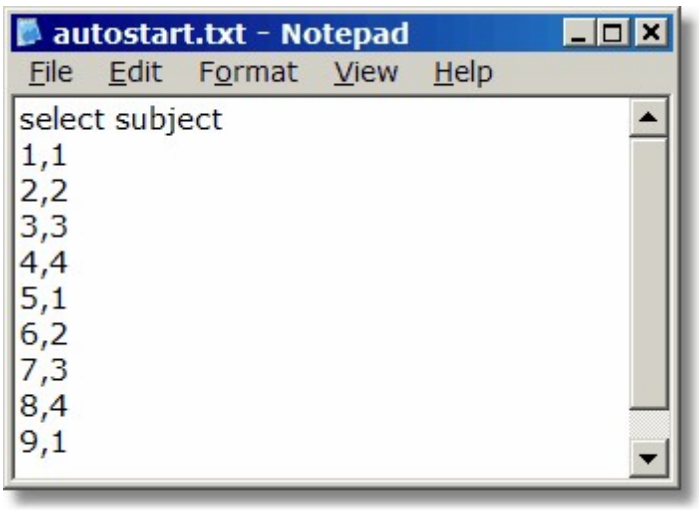
By default, each time you run a MediaLab experiment, you need to enter a subject ID and condition. You can automate this process by using an AutoStart file. All you have to do is generate a text list of subject IDs and conditions and call it "autostart.txt" and place it in your experiment folder. For example:



When you run the experiment, MediaLab checks to see if an autostart.txt file exists in the same folder. If it does exist then it automatically runs the next subject ID and condition on the list. If it doesn't then you will be prompted for the subject ID and condition as usual.

MediaLab will mark the subject ID condition with a * so it knows where to start next time. Following the session, MediaLab will ask if you want to run the next session--if you say yes it automatically gets the next subject ID and condition from the list and runs it. This way you can run sessions all day on multiple computers and never have to enter a subject ID or condition. Note that you will probably want to create a unique list of subject ID's and conditions for each computer to prevent redundancy!

Finally, an alternative way of using the AutoStart feature is to type "select subject" as the first line in the autostart.txt file. If you do this, then MediaLab will prompt you only for the subject ID. It will then look up the condition for that subject ID from the list and run it automatically. This is useful if you want to have a constant autostart.txt file on all of your computers and don't mind keeping track of which subjects still need to be run. It simply eliminates the need to keep track of which conditions to assign.



Advanced Hints for AutoStart

Using a *substitute.txt* 86 file, you can have a blank message ("") defined for the end of session event. If MediaLab sees that you have no message *and* you are using an autostart file, then it will proceed automatically with the next session without asking if you want the next session run; it will simply start. Doing so requires ctrl-alt-delete when you finally want to escape.

Note about ending sessions early:

The *autostart.txt* file will only be updated if a session is *completed*. You can end a session in two ways. One is to press *ctrl-alt-delete* and choosing to shutdown the MediaILb program--this will shutdown the session, no data will be saved, and the autostart file will remain unchanged. The other is to press *ctrl-alt-right,* which is the key combination to end a session. If you end a session in this manner the data *will* be written including missing values for questions that were never answered and you will be asked if you would like the autostart file to be reset as it was before the session. Answering *yes* to this will prevent MediaLab from scratching this session off its to-do list.

Overview

MediaLab allows you to create your own custom items using HTML. Using the *Custom* item type, you can specify any web browser file (local or internet) in the *Question Wording/File Name* field. If and when a *post* event occurs, MediaLab will save any submitted data that you like and then continue on to the next item in the questionnaire. In order for a posted variable to be saved with the regular data files, its variable name simply needs to match that of the *Custom* item or any other item in the current experiment.

HTML Items vs Custom Items

An *HTML file* (e.g., *mydoc.htm*) is a document. In the same way that a *.doc* file is meant to be viewed in *Microsoft Word*, an *.htm* file is meant to be viewed in a web browser like *Internet Explorer, Chrome,* or *Firefox*. An *.htm* file is also a simple text file which means you can open and edit it in any text editor such as the Windows *Notepad*. When you open an *.htm* file in a text editor you will see code that you don't see when viewing it in a web browser--these are all the instructions that tell the web browser what to present and

what to do with user actions.

Both *Custom* and *HTML* items in MediaLab may contain pretty much any HTML code you like. This means that anything that can be displayed in Internet Explorer, whether it's on the web or on your local computer, can also be used. This can include pictures, movies, formatted text, hyperlinks, flash animations, interactive java applets, scripts, and so on. However, there is a limit to the *HTML item type*--you can't collect any data with it. You can *include* questions but there has never been a way to get MediaLab to save those data *in the regular data files*. Contrariwise, the *Custom* item type allows you to do just that through the use of standard HTML *form code*.

Note that a useful introduction to HTML is beyond the scope of this documentation. For a general introduction see:

www.w3schools.com/html/html_intro.asp
www.w3schools.com/html/html_examples.asp

What is an HTML Form?

Forms are the basic ingredient for asking questions on the Internet. A single question or set of questions can be asked in any HTML document by surrounding some question code (discussed shortly) with the *<form>* tag like this:

```
<form method="post">
    questions and variable names for responses go here
</form>
```

The *<form method="post">* tag tells the browser to keep track of any data that is collected within the form and make it available for saving when the user clicks a button to submit their responses. Although it can get more complicated, this is the gist of how most questions are asked when you are surfing the internet.

For more information regarding HTML forms, see:

www.w3schools.com/html/html_forms.asp

How it Works - MediaLab Intercepts!

If you present a Custom item rather than an HTML item, then MediaLab will monitor the posting of any form data. If it finds any data being submitted where the variable name *matches* the name of any item in any questionnaire file in your MediaLab experiment, then that intercepted value will be written to your standard data files under that name. That's all there is to it.

Example

What is the mysterious code that goes between the *<form>* tags? This item below presents a text box in which subjects can enter a response. When they click *Continue*, MediaLab will check to see if any variables in the experiment are called *textBox1*. If so, the response will be saved to the data files under that name. We can handle that easily by assigning the variable name *textBox1* to the Custom item itself when we list it in the questionnaire.

```
<form method="post">
    What is your answer?
    <input name="textbox1">
    <input type="submit" value="Continue">
</form>
```

How it looks in a browser:

Try It!

To create your first custom item you can copy the code above (everything from *<form>* to *</form>*) into a blank text file (e.g., via Notepad). Save it somewhere as *myTextBox.htm.* Voila! You have just created a custom item! You can view it in Internet Explorer if you like by double clicking on it.

To use it with MediaLab, open a questionnaire file and add an item--choosing *Custom* as the item type, *myTextBox.htm* as the file name, and *textBox1* as the variable name. That's it! Try running the questionnaire and then view your data to see if the posted response was written to your data file. Remember that if you have made changes that will affect your data files, you'll want to click the *Clear Data* 75 button to delete the old data files and start fresh (after backing them up if there are data you want to save).

Hint: You can see these samples already prepared in the *C:\MediaLab\Samples\Sample 5 Custom Items* folder.

Advanced Hint: Colours, fonts, & focus

Adding a few lines to your HTML file can make life a little easier. The additions are in black:

```
<!-- saved from url=(0013)about:internet -->
<style><ml.style></style>
<body onload="document.form.textbox1.focus();">
  <form name="form" method="post">
    What is your answer?
    <input name="textbox1">
    <input type="submit" value="Continue">
  </form>
</body>
```

The first line allows you to preview the HTML file in Internet Explorer without any kind of warnings popping up. The second asks MediaLab to apply the colors and fonts used in the current session. The third line requests that the screen cursor appear within the specified input area. In this example, we are asking that the cursor automatically appear within the *textbox1* field so that the users does not need to click on it to start typing. To try the example above, see *C:\MediaLab\Samples\Sample5 custom items\myTextBox2*.

Advanced Hint: Continuing without a Continue Button.

MediaLab will not display the standard *Continue* button when a custom item is used. Instead, it will wait until some data is posted via a submit button. In some cases though you may not want this to happen automatically. An alternative is to use the password parameter 61. In this case the password can be a variable name, e.g., *(xvarname)*. If you do this, MediaLab will not proceed to the next item unless the password variable is included in the post. This is especially handy if you want to make a multi-page custom item with multiple postings.

See Also

Repeating Custom Items 91

Single Vs. Multiple Variables 92
Variable Reference 94
Samples 77

9.1 Repeating Custom Items

A powerful and time-saving feature of custom items is the capability to use them over and over again without having to modify them. If you create a special scale response item with a slider to use in your 50-item questionnaire, you do not want to make 50 versions of it, each with different question wording and response options. Well, maybe you do. If so, no need to read any further. But if you *like* the idea of 1 file for 50 questions, read on!

Instead of adding the exact question wording to your custom item's HTML code, you can simply say *<ml.wording>* and MediaLab will automatically *substitute* the question wording from your questionnaire file. This means, each time you use your custom item in a MediaLab questionnaire, you can enter different question wording the you would for any other normal item. MediaLab will take it and put it into you HTML file for you!

Likewise, instead of using a specific variable name in the HTML form code, you can simply specify *<ml.varname>*. This means you can now use the *same custom item file* with many questions in the same questionnaire because each time, MediaLab will substitute the current variable name into the HTML.

You can also substitute other values such as backgrounds, backsounds, backvideos, fonts and colors from the current session and so forth without having to change your item. For all the stuff that MediaLab can substitute into your HTML, see the variable reference 94 table.

Note it is super supremely critical to use the *<ml.varname>* substitution if you are going to be using the same custom item file over and over again in the same experiment. Otherwise, the posted variable name will be always be the same and so the response will always be written to the same variable in the data file. All of the other substitutions can provide for some handy tricks but they are never necessary.

Substituting Values from the MediaLab Session

In the case of our custom example above, try this modification and save it as *myTextBox3.htm*

```
<form method="post">
    <ml.wording>
    <input name="<ml.varname>">
    <input type="submit" value="Continue"">
</form>
```

Create three *Custom* items in your questionnaire and assign them variable names such as *ques1*, *ques2*, and *ques3.* For each, enter the same custom item file: *myTextBox3.htm* for the file name. As you do this, create a different question wording for each one (e.g., *What is your favorite food? What is your favorite animal? What is your favorite movie?).* Now run the questionnaire and see what happens. You should find that each time you request it, MediaLab will display it with the appropriate wording from the questionnaire and will save the data from the item using the appropriate variable name.

See Also
Custom Items, Overview 88
Single Vs. Multiple Variables 92
Variable Reference 94
Samples 97

9.2 Single Vs. Multiple Variables

Single vs. Multiple Variables

The previous example item 88 collects a single response. The execution of this method is the easiest and is applicable when you have a single value being collected with your custom item. In your MediaLab questionnaire, you name your custom item *var01* and in your HTML form, you likewise name your posted variable *var01*. When the value for *var01* is posted, MediaLab will intercept it and save it as the value for *var01* in your data files. Done deal. It's a little trickier when you have multiple variables being collected with a single custom item. To deal with this, you have two choices--use a *variables* parameter or create *dummy* variables in your experiment.

Using a variables Parameter

To use the *variables* parameter, add a *v* parameter to your *Custom* item, indicating the *number of variables* you want to collect, e.g., *(v7)* would create space for seven variables. The variables will be named after the primary variable name with an underscore and numbers appended. So for example, if the variable name of your *Custom* item were *choice*, you could post variables in your HTML form called *choice_01*, *choice_02*, ... *choice_07*.

In this example, we ask the participant to list "three good movies". It's very similar to the first example, but here we create three blanks instead of just one. In this case we'll need three variables so we will add the parameter *(v3)* to our *Custom* item that we are calling *movies*. Using the *v* parameter means that we will reference the variable names by using the primary variable name (*movies*) along with *_01*, *_02*, and *_03*.

```
<form name="form" method="post">
   <ml.wording>:<br><br>
   <input name="<ml.varname>_01"><br>
   <input name="<ml.varname>_02"><br>
   <input name="<ml.varname>_03"><br><br>
   <input type="submit" value="Continue">
</form>
```

How it looks in a browser:

To see this sample is action, go to *C:\MediaLab\Samples\Sample5 custom items \myTextBox4 - multiple variables*

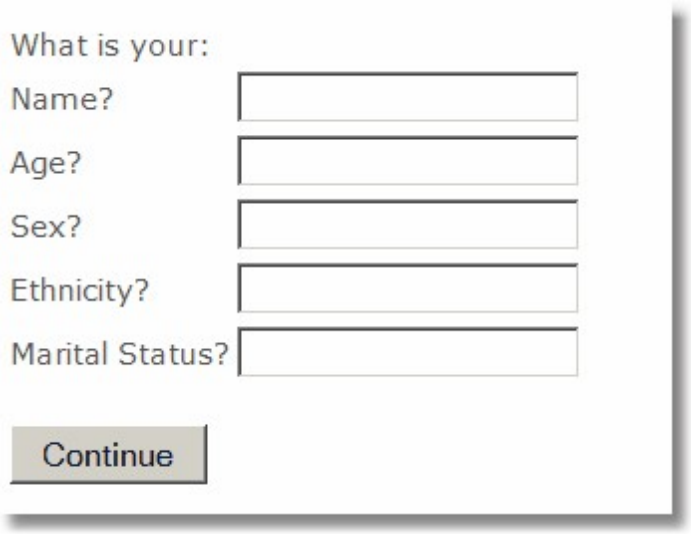
An alternative: Dummy Variables

Instead of adding a parameter, you can create dummy variables in your experiment that act as placeholders for data with matching variable names. To do so, create an additional *Custom* item in any questionnaire and give it whatever variable name you like but do *not* give it a *filename*. When MediaLab sees that you have listed a *Custom* item with no file name indicated, it will understand that this item is merely a *placeholder* for a variable being collected in some other *Custom* item. When recognized as a placeholder or dummy variable, such items will be skipped during the experimental session.

This method provides a nice way of capturing data from questions or surveys where you can not control the variable names that are used in the existing HTML forms. For example, if you look at the source code for a survey on the internet and see that you want to save two variables called *age* and *sex* from the many that are being posted, you can include the survey as a *Custom* item and then add *age* and *sex* as two dummy variables in your experiment. In the following example code, the *<table>* tag along with cell markers are used in order to maintain an orderly look via rows (*<tr>*) and columns (*<td>*).

```
<form method="post">
<table>
    <tr><td> What is your: </td></tr>
    <tr><td> Name? </td><td> <input name="name"></td></tr>
    <tr><td> Age? </td><td> <input name="age"></td></tr>
    <tr><td> Sex? </td><td> <input name="sex"></td></tr>
    <tr><td> Ethnicity? </td><td> <input name="ethnicity"></td></tr>
    <tr><td> Marital Status? </td><td> <input name="marital"></td></
tr>
    <tr><td> <br><input type="submit" value="Continue"></td></tr>
</table>
</form>
```

How it looks in a browser:



To see this sample is action, go to *C:\MediaLab\Samples\Sample5 custom items \myTextBox4 - dummy variables*

See Also
Custom Items, Overview 88
Repeating Custom Items 91
Variable Reference 94
Samples 97

9.3 Variable Reference

Variable Reference

The following variables can be placed in the HTML of any custom item. MediaLab will substitute the appropriate values in the HTML code before it is displayed. This is a convenient way to substitute information from the current MediaLab item without having to alter the HTML code. Such substitutions could include color and font information, response options, question wording, background images, parameters, and so forth.

Note that you can also substitute calculated values from a responses.xls file 101 by placing the variable name in angular brackets. For example, if you had a calculated value named "total" in your *responses.xls* file, you could enter *<total>* in your HTML file and MediaLab would substitute the correct value at the time the HTML is displayed.

<ml.subject>	Subject ID	**<ml.$>**	quickstyle
<ml.condition>	Condition	**<ml.@>**	time stamp
<ml.varname>	Variable Name	**<ml.c>**	command line arguments
<ml.wording>	Question Wording. Will also include substitutions	**<ml.d>**	duration - *not available as a variable*

	from *responses.xls* if applicable.		
<ml.color>	Main area text color. Will have prefix "#" e.g., *#CCCCFF*	**<ml.h>**	height
<ml.bgcolor>	Main area background color. Will have prefix "#" e.g., *#CCCCFF*	**<ml.k>**	key
<ml.font>	Main area font name, e.g., *arial*	**<ml.l>**	left position
<ml.fontsize>	Main area font size. Will have "pt" appended, e.g., *16pt.*	**<ml.m>**	mask
<ml.color2>	Input area text color. Will have prefix "#" e.g., *#CCCCFF*	**<ml.n>**	number of thoughts to take
<ml.bgcolor2>	Input area background color. Will have prefix "#" e.g., *#CCCCFF*	**<ml.o>**	question wording onset
<ml.font2>	Input area font name, e.g., *arial*	**<ml.p>**	number of scale points (1-12)
<ml.fontsize>	Input area font size. Will have "pt" appended, e.g., *16pt.*	**<ml.q>**	target question for thought rating
<ml.bg>	BackGround. Append parameters if desired: .t, .l, .h , .w, e.g., <ml.bg.w>	**<ml.r>**	range
<ml.bs>	BackSound	**<ml.s>**	spacing
<ml.bv>	BackVideo. Append parameters if desired: .t, .l, .h , .w, e.g., <ml.bg.w>	**<ml.t>**	top position
<ml.text*x*>	Text Label (1-12). Replace *x* with option number, e.g., *<ml.text7>*	**<ml.ttl>**	send ttl signal

<ml.label*x*>	Button Label (1-12). Replace *x* with option number, e.g., *<ml.label7>*	**<ml.w>**	width
<ml.skipto*x*>	SkipTo Value (1-12). Replace *x* with option number, e.g., *<ml.skipto10>*	**<ml.x>**	password required to continue
<ml.goback>	Whether GoBack is allowed. True/ False.		
<...>	Any variable from responses.xls. Specify variable in angular brackets, e.g., *<var1>*.		

All of the variables above may be directly substituted into your HTML file. For example, if the question wording in you HTML were: "Please choose from one of the <ml.p> alternatives" and you had used the parameter (p5), the question wording would appear as "Please choose from one of the 5 alternatives."

The code to display an image in HTML may look like this:

```
<img src="mypic.jpg">
```

Instead of having this be a fixed image, you could have your custom item pull up a different picture every time by placing an image in the *background* field in the questionnaire. Then, in your HTML you could specify:

```
<img src="<ml.bg>">
```

Voila--a different picture every time--whatever you have put in the BackGround field!

Indirect Substitution (optional)

Values may be directly substituted by entering them directly into the HTML code as above. This can make previewing the HTML (e.e., in Internet Explorer) a little difficult because IE won't know what <ml.bg> means. If you don't care about previewing the HTML, then it doesn't really matter. However, there is an alternative method called *indirect substitution* where you can list the substitutions *below* all the HTML code like this:

```
<form method="post">
     <img src="mypic.jpg"><br>
     some wording here
    <input name="varname">
    <input type="submit" value="Continue"">
</form>
```

```
<!--ml.subs
ml.sub "question wording" = <ml.wording>
ml.sub "varname" = <ml.varname>
ml.sub "mypic.jpg" = <ml.bg>
-->
```

See Also
Custom Items, Overview 88
Repeating Custom Items 91
Single Vs. Multiple Variables 92
Samples 97

9.4 Samples

Some Custom Sample Code

Examples of custom items and various code snippets can be found throughout the forums at www.empirisoft.com/support

The following examples of HTML form code are taken from www.w3schools.com/html/html_forms.asp

Text Fields

Text fields are used when you want the user to type letters, numbers, etc. in a form.

```
<form>
First name: <input type="text" name="firstname"><br>
Last name: <input type="text" name="lastname">
</form>
```

How it looks in a browser:

Note that the form itself is not visible. Also note that in most browsers, the width of the text field is 20 characters by default.

Radio Buttons

Radio Buttons are used when you want the user to select one of a limited number of choices.

```
<form>
<input type="radio" name="sex" value="male"> Male <br>
<input type="radio" name="sex" value="female"> Female
</form>
```

How it looks in a browser:

Note that only one option can be chosen.

Checkboxes

Checkboxes are used when you want the user to select one or more options of a limited number of choices.

```
<form>
<input type="checkbox" name="bike">I have a bike <br>
<input type="checkbox" name="car">I have a car
</form>
```

How it looks in a browser:

The Submit Button

When the user clicks on the "Submit" button, the content of the form is sent to another file. The form's action attribute defines the name of the file to send the content to. The file defined in the action attribute usually does something with the received input.

```
<form method="post">
Username: <input type="text" name="user">
<input type="submit" value="submit">
</form>
```

How it looks in a browser:

If you type some characters in the text field above, and click the "Submit" button, you will make the text data available to MediaLab for the variable User.

Source: www.w3schools.com/html/html_forms.asp

Back to Custom Items Overview 88

Responses.xls: Calculated Values, Complex Skips & Reports

Overview

This section will show you how to accomplish three advanced functions with MediaLab:

- Calculate scores and other variables while a session is running. Present these calculated values on screen as stimuli or feedback, use them in skip patterns, or in post-session reports.
- Execute complex skip patterns. Base skips on any prior response, or even combinations of prior responses and calculated variables.
- Create and optionally print summary reports immediately following the experimental session. Create graphs, scale scores, or anything else you want based on raw and/or summarized data.

Sample

Before we look into the technical details of accomplishing these advanced functions, let's run a quick sample that demonstrates all three. Note that to take advantage of the functions described in this section, you will need to have Microsoft Excel installed on your system—both to design and to run the sessions. Also, because these are considered advanced features, we're going to presume you have some basic familiarity with MediaLab.

From the samples folder, run Condition 1 of the following experiment:

```
C:\MediaLab\Samples\Sample5 advanced features\advanced.exp
```

After the first few questions, this should appear to look just like the standard self-esteem questionnaire sample from the introductory tutorial.

What is very different from the standard tutorial is the summary screen which follows the self-esteem scale. Somehow, we have been able to calculate your overall self-esteem score and present that value back to you. We have also been able to tell you what that score means—i.e., whether you are high or low in self-esteem—complete with matching emoticon—i.e., :) or :(.

We have also been able to calculate your average response time and tell you if you've been fast or slow. On the following screen we continue by asking a question about your membership in a racial minority or majority—based on how you responded to the race question earlier in the questionnaire.

What happens next will depend on how you scored on the self-esteem test. You can try this a couple times to see for yourself. If you scored moderately (2, 3, or 4), then you will be at the end of the session. If you scored very high (greater than 4) then you will be at a screen representing where a social desirability scale might be administered. If you scored very low (less than 2) then you will be at a screen representing where a depression inventory might be administered.

When you're done, go to the experiment's data folder. In here, you'll find a folder called "reports". Inside the reports folder is a file called *1.xls* (this will be different if you used another subject ID). Open this file by double-clicking on it. First you'll see many of the responses you gave to the questionnaire. Click on the tab at the bottom left that says "sheet 2". Here you'll find a custom report for the session that displays your name, age and self-esteem score. It presents a graphic that is based on your self-esteem score and also displays a graph of the individual response times for each question. This report is completely customizable and can be created automatically for each subject—with any raw

or summary data you like, and formatted in any way you like.

How is it all done?

Let's close the report file and go back to the experiment folder. Open the file *responses.xls*—this is the secret. If you have a standard Excel file called *responses.xls* in your experiment folder then MediaLab knows you want to use some or all of these advanced functions. The question now—what do we put in this Excel file called *responses.xls*?

Variable, Value and Skipto

Notice that there are three columns on the opening worksheet—variable, value, and skipto.

The Variable Column

In the variable column there are two types of variables—experiment variables and calculated variables. If you take a look in our questionnaire self-esteem.que you'll see that many of the variables listed there are also listed here in the variable column of *responses.xls*—name, age, rse1, rse2, etc. These are experiment variables—i.e., variables taken from any of our questionnaire files in the experiment. This also includes reaction times such as trse1, trse2, trse3 and so on. If it would normally appear in your regular MediaLab data files, then it can be used as an experiment variable in the *responses.xls* file.

But what about variables like minmaj, rse (positive scored), rse (positive scored), and rse? These are not variables from the experiment. That's because they are calculated variables we are creating them here in the responses.xls file. You can create any and as many new calculated variables as you like and you can call them whatever you like. The only rule for a calculated variable is that it has a name that isn't used in your experiment.

The Value Column

This brings us to the value column. For experiment variables, this is really simple—you don't need to do anything. As a subject progresses through your experiment, MediaLab will automatically fill these values in with whatever value is recorded for that variable in the regular data files. After every question in your experiment, MediaLab checks the variable column of the *responses.xls* file to see if it can fill in any values with the newly acquired data. Thus, experiment variables are automatically updated by MediaLab when the data become available.

What happens next is the really fun part–Excel automatically updates all of your calculated variables taking into account the current values of your experiment variables!

Let's look at the example of rse (positive scored) . Double-click the mouse on the value column for this calculated variable. You'll see that this cell actually contains the formula:

```
=AVERAGE(B4,B5,B7,B9,B10)
```

This is an Excel calculation formula. We're telling Excel that we want the value here to equal the average of the cells listed in parentheses. If you look at what's in those cells, you'll see that it's the subject's responses to *rse1*, *rse2*, *rse4*, *rse6*, and *rse7*. These happen to be the positively scored self-esteem items such as "I feel that I have a number of good qualities."

Now let's look at rse (negative scored) . You'll see that this cell contains:

```
=7-AVERAGE(B6,B8,B11,B12,B13)
```

Here we're averaging the reverse scored items (e.g., "I am inclined to feel that I am a failure.") and then subtracting that value from 7 to recode it such that higher numbers reflect higher self-esteem.

Finally, look at rse :

```
=AVERAGE(B27,B28)
```

We see that the final calculated self-esteem score is an average of the two sub-scores above. Now we have a calculated variable rse that is the subject's overall self-esteem score.

10.1 Calculated Scores

Presenting Calculated Scores in MediaLab

Now that we have calculated a number of variables from the current session, we can present these values back to the subject. Take a look at item #15 in the self-esteem questionnaire:

```
You have <rselabel> self esteem <face>

You scored <rse> out of a possible 6.

Your average response time was <rsert>ms (<rsertspeed>).
```

We are presenting six calculated variables here—all from the *responses.xls* file. On the second line of the above question wording we see where we are using the calculated self-esteem score. We do this by placing the name of the calculated variable in angular brackets like this: *<rse>*. That's all there is to it—using this method, you can insert the value any calculated variable into the question wording of any MediaLab item.

Following the same idea, we calculated and presented whether the subject was high or low in self-esteem by using the calculation for <rselabel> as follows:

```
=IF(B29>3,"high","low")
```

Using Excel's language for conditional logic, here we're saying that if the subject's calculated self-esteem is greater than 3, then they're "high" and that otherwise, they're "low". Calculating and presenting the subject's average response time, whether it is considered "slow" or "fast" are additional examples of the same idea—calculating new variables from gathered data and using those values in the same session.

Save Calculated Variables to Your Data Files

You can save calculated values along with your other MediaLab data. Create simple placeholder variables in your questionnaire by using the *Custom* item type with *no filename*. When such an item is encountered, MediaLab will check to see if the variable name exists in your *responses.xls* file. If it does, then MediaLab will record the value for that variable in your regular data files. In other words, by using the same variable name for the placeholder and calculated variables, MediaLab will know which values to grab from responses.xls and store in your regular data files. Note that MediaLab will be grabbing the current value of the calculated variable if and whenever it comes across the placeholder item during the session--i.e., location of the placeholder within your session could be an important factor.

See Also: Advanced Features Overview 99

10.2 Complex Skips

Just as calculated variables can be used as information we can present back to subjects, they can also be used to help us with skip logic. For any experiment variable you have entered in the *reponses.xls* file, you can specify or calculate a variable to skip to when that item is completed. If the current MediaLab item is listed in the variable column of the *responses.xls* file, then MediaLab will check the *skipto* column when the item is finished to see where it should go next. If there is no *skipto* value then no skip will occur.

Let's take the fill-in-the-blank age question for example. Currently, there is no *skipto* value. What would happen if we typed "debrief" into the *skipto* column for that item? When the age item came up in the MediaLab session, MediaLab would see that age is listed in the *responses.xls* file. It would see the *skipto* value of *debrief* and send the subject to the last item in the questionnaire because it is named *debrief*. If you have used MediaLab's *skipto* function in the past, this is pretty standard stuff.

Where it gets interesting is in the fact that you can now use the *responses.xls* file to calculate *skipto* values. Instead of placing a static value in the *skipto* column, we can compute a variable based on what's been happening in the experiment. Let's say that we want to use the subject's self-esteem score to determine what happens next. If they scored extremely high, we want to administer a scale which measures socially desirable responding. If they scored extremely low then we want to administer a depression inventory. If they scored in the normal range, we'll skip them to the end of the session.

Take a look at how we do this by checking out the formula in C26, that is the *skipto* value of the essay item, *ess1*:

```
=IF(B29<2,"dep",IF(B29>4,"sdes","debrief"))
```

In this formula, you can see three items from the questionnaire file—*dep*, *sdes*, and *debrief*. These are our three *skipto* candidates. The conditional logic of Excel is applied to say: "If self-esteem is < 2 then dep otherwise if self-esteem is > 4 then sdes, otherwise debrief."

See Also: Advanced Features Overview 99

10.3 Post-Session Reports

At the end of every session, MediaLab automatically saves a copy of Responses.xls in the data\reports folder. The copy is named using the current subject ID and will have all of the subject's responses to the experiment variables saved in the value column. There are two reasons for this. One is that this allows you to open the subject's report and check your calculations. You will be able to view the results of all of calculated variables and skipto values based on the subject's actual responses. It's a handy way to check your work.

The second reason for this is the option to create post-session reports for each subject. In the sample provided, you can click on the "sheet2" tab in the lower left corner of Responses.xls. You will see a template for a post-session report. This is a completely free-format. It uses Excel formatting to create text and figures with references to experiment and calculated variables on the main worksheet. As the session unfolds and

values are updated on the main sheet—so are the fields and figures in the report. When the session is over, the report is saved for easy access to these summary data. Take a look at "sheet2" of data\reports\1.xls as an example.

Notice that on the main page of responses.xls there are two lines that read:

```
print.sheet1  no
print.sheet2  no
```

You can add these optional lines to any responses.xls file. If listed with "yes" in the value column, they instruct MediaLab to send the listed worksheet(s) directly to your default printer. This is handy, for example, if you want a report on "sheet2" to print automatically so that you might discuss the results with the participant.

See Also: Advanced Features Overview 99

10.4 Tips

Extra Notes & Rules to Know

- When creating a *responses.xls* file and saving it for the first time, the File name is just the word "responses" (without the quotation marks) and the type of file must be .xls. Any use of a *responses.xls* must follow this convention and must be saved in the same folder as the MediaLab files that use it.
- As a guiding principle remember that MediaLab never looks at your formulas—only the results! How the results of calculated variables occur is completely between you and Excel. Remember that it's your recipe–MediaLab just supplies the ingredients and eats the meal.
- You can include as many or as few experimental variables as you like; you don't need to list every variable from your questionnaire files in the *responses.xls* file. Feel free to include only those that you need for your calculations.
- A complete discussion of Excel's logic and math functions is way out of the scope if this tutorial. If you've never done calculations in Excel this might seem a bit intimidating at first, but it's really not too bad (honest!). The trick is getting an understanding of how to nest multiple conditions within a single calculation. Grab a pot of coffee and go through Excel's help on this for an hour or so and you'll get it.
- To access the relevant help on constructing calculations and conditional statements, press F1 in Excel. Using the Answer Wizard or the Index, look up terms like If, And, Or, Logical Functions, Functions Listed by Category.
- When including experiment variables in the *responses.xls* file, you can include ANY variable from ANY questionnaire in your experiment—everything goes into the *responses.xls* file, which is placed in your experiment folder.
- Variable names used for experiment variables need to match the variable names used in the MediaLab data files. This is important to remember for items that record multiple values. For example, let's say you have a multiple response question called MR1 that has five response options. Normally, there won't be a variable in your data file called MR1. Since there are five responses written, they will be called MR1_01, MR1_02, MR1_03 and so forth. This applies also to thought listing items and ranking items. If in doubt, you can always check one of your data files to see how the variable names are written.

- When creating conditional statements or calculations based on the value of experiment variables, it will always be the value written to the final data file that MediaLab will enter in the value column of Responses.xls.
- Essay content can not be accessed as a variable.
- *responses.xls* files should not have any empty rows on the main page
- MediaLab executes skips by skipping over items until the desired item is found. This means that if you tell MediaLab to skip to a variable that doesn't exist then the session will end.
- Using a *responses.xls* file will have no impact on your primary data files—i.e., calculated variables are just for use during the experiment and are saved only in the post-session report (i.e., in the *reports* folder see above).
- In addition to saving a copy of the *responses.xls* file under the subject's ID in the data \reports folder, MediaLab will also save a copy as !currentsession.xls in the same folder. This is a temporary file reflecting the responses of the most recent participant in the experiment.
- Currently the MediaLab *GoBack* function is disabled if calculated skips are involved..

See Also: Advanced Features Overview 99

QuickInfo

Here are a few miscellaneous tips that don't easily fit elsewhere in this guide.

Specifying File Paths 104

Sizing Images and Movies 105

Miscellaneous Features 106

Using Responses from MediaLab as Stimuli in DirectRT 106
Adjusting Speaker Volume Automatically During Session 107
Secondary Tasks - Requiring Participants to Respond to Probes 107
Sending and Receiving Serial Data 107

Trouble Shooting Tips 109

11.1 Specifying File Paths

When you tell MediaLab to present a file (e.g., an image, a sound, a movie, a questionnaire, etc.), you need to specify where that file is located. There are two ways to do this—using *absolute* or *relative* paths.

Absolute Paths

If the file you are presenting exists anywhere outside of the experiment folder then you will have to specify an absolute file path. An absolute file path is the exact location of the

file (e.g., *c:\mypictures\myimage.bmp*). You can find the absolute pathway of a file by right-clicking it and selecting Properties. Next to the word "Location:" in the General tab of the window that opens, you will find the absolute pathway with everything except the file name at the end. Simply copy the entire pathway next to the word "Location:" paste it where you need it, and be sure to add "*\your file name here*" (without the quotation marks) to the end of the pathway you copied.

Remember, if the file is not in the same directory as the experiment or it's not in a subfolder in the experiment file folder, then you must specify the complete path of the file. Make sure that any absolute file paths you use will still be valid if you copy your experiment to another computer.

Relative Paths

If the file is located in the same folder as your experiment file, then you can simply enter just the name of the file (e.g., *myimage.bmp*). If the file is located in a subfolder that is located in the same folder as your experiment files, then you can enter the name of the subfolder followed by the name of the file (e.g., *images\myimage.bmp*). You can use a relative path for ANY file located anywhere within your main experiment folder.

The advantage of placing the file in the experiment directory (or a subfolder) is that the experiment folder can then be moved to a different place and you won't have to worry about checking path names (e.g., *c:*\..., *d:*\..., etc.). You could copy your experiment folder to any location on any drive and all of the relative paths would still be valid.

Hint

You can specify relative or absolute file paths to be the default when selecting files in the Experiment Editor. You can set your preferred default in the Options menu 15. It is recommended to set this option to *Use Relative Paths*. This is because the only time an absolute path is necessary is when the desired file is not located somewhere within the experiment folder. However, even if you are using relative paths, the editor will enter an absolute path if this is the case.

11.2 Sizing Images and Movies

MediaLab tries to use a resolution-independent scale so that your experiments will look the same no matter what resolution of the computer on which you run them. Some differences can not be avoided however. To see what the scale is on your particular system, choose "Show Location Points" from the Help menu in MediaLab. To see how the resolution independence works, take a look at the scale now. Then resize the MediaLab window and try it again. You should notice that the scale doesn't change. Try setting a different display resolution on your computer (Control Panel -> Display, Settings). Look at the scale again—it should be about the same no matter how you change your display. One factor that does to to influence the scale is whether you use "small fonts" or "large fonts" in your system settings. Small fonts produce a slightly larger range for both height and width. If you want the experiment to look the same on all computers you should check to make sure they are all running small fonts or that they are all running large fonts.

Typically, the width parameter 63 will range from 1 to about 700. The height parameter 63 will range from 1 to about 500. These values apply to the top and left parameters 60 as well.

Note that the scale applies to most everything in MediaLab when the top, left, height, and width parameters are used. The exception is movies. Movies do not follow the MediaLab scale because they are presented with the Windows MediaPlayer. Consequently, if you specify a width of 300 (for example) it will appear smaller on a larger

screen resolution. How you can combat this is to use the special width parameters 63 provided especially for movies. For example, (w-1) will play the movie at full screen, (w-2) will play it half screen, (w-3) at a quarter and there are others.

Note also that although the scale applies to images, that this is only true if you explictly define the size of the image (using the height and width parameters). Otherwise the image will appear as it's default size. For example a 300 pixel wide image will appear smaller when you use higher screen resolutions. However if you explicitly define the size using the height and width parameters then it will appear according to the MediaLab scaling system. You can find more information about this in the following sections of the manual:

Parameters (Questionnaire Files) 57

Parameters (Experiment Files) 31

The most common resolutions are 640x480, 800x600, and 1024x768. The first number tells you the number of pixels that are displayed across your screen, and the second number tells you the number that are displayed down your screen. You can determine and/or set the resolution of your display by right clicking on your desktop and selecting *Properties* > *Display*. MediaLab tries to use a resolution-independent scale so that your experiments will look the same no matter what resolution you run in. Some differences can not be avoided however. To see what the scale is on your particular system, choose *Show Location Points* from the *Help* menu in MediaLab.

By right clicking on your desktop and choosing Settings you can see if your system is set to use large or small fonts. In 640x480 mode, only small fonts are available. At greater resolutions, you may optionally choose large fonts. This impacts on window sizes as well. Larger fonts result in larger windows. In our testing, MediaLab has worked well in most resolutions with both large and small fonts.

11.3 Miscellaneous Features

These features are documented here despite the fact that not as commonly used as the others discussed in this manual. Sometimes when people get customizations to MediaLab done, they are very useful to a lot of other users and so they get worked into the manual. Other times, well… they'll probably gather some dust. But just in case somebody else might be able to benefit from them, here they are:

Using Responses from MediaLab as Stimuli in DirectRT

At some point you may want to use a subject's answer to a scale response 50 or fill-in-the-blank 44 item as a stimulus in DirectRT. For example, in MediaLab you could ask a series of fill-in-the-blank items asking subjects to enter foods that they like. Or you might ask them to indicate their race using a scale response. Then in a DirectRT session, you might want to display these responses as stimuli (e.g., as primes or targets in a priming study). For fill-in-the-blanks and scale responses, you can specify a file name ending with .txt in the *File Name* 41 field to have MediaLab save the response in the file you specify. This will cause MediaLab to save the subject's response preceded with a "~" symbol so that DirectRT will recognize that it is text for display purposes. Then, in DirectRT, you can specify that this text be displayed as a stimulus by using the "&" symbol (which tells DirectRT to read a single line stimulus from a text file). So if you have a MediaLab question that asks subjects their race called "race" you could enter something like "c:\race prime study\stim\race.txt" in the *File Name* field. Then, in your DirectRT input file, you could refer to "&race" as a stimulus. When DirectRT finds this, it will open the race.txt file that MediaLab created, and will present the subject's response as the stimulus. See the DirectRT documentation regarding "Input Files" and "Stim" types for more detail.

Adjusting Speaker Volume Automatically During Session

Sometimes you will have multiple media clips in your session and they vary in volume. Some play too loud, others not loud enough. Now you can embed an executable item at any point called *volume.exe* located in the *C:\MediaLab\Utilities\Sounds* folder. Copy this file into your experiment folder. Anywhere in your experiment add an executable item and in the F*ile Name* field, type *volume.exe*. In the parameters field enter *(c5)* where 5 is a volume setting from 1 to 10. For max volume, enter *(c10)* and for min volume enter *(c1)*. Any integer value from 1 to 10 is permitted. You'll get a brief, "please wait one moment" message while the volume is adjusted, after which you can click to continue.

Secondary Tasks - Requiring Participants to Respond to Probes

Visual, auditory and TTL signal probes can be launched at any time from your questionnaire. You can create a schedule of probes at either random or fixed intervals and get subjects' response times to the probes. The probe schedule can span multiple items as it runs independently of the experiment once launched. Response times are assessed using DirectX so their accuracy is very good.

To execute a probe task you create a schedule of probes in an Excel file. Each schedule may contain multiple sets of probes, with each set having an ID such as 1, 2, 3 and so on. In a MediaLab questionnaire, we refer to this ID in the parameters of any item by saying "*1" or "*2". That tells the Probe Generator which set of probes to execute. The set will execute until completion which means that a single set can span one or many items in MediaLab. You can have as many sets of probes as you like (Set 1, Set 2, Set 3 and so on). However, you can only request a given set once in a session. You might assign different probe sets to different tasks within the same session, or to the same task in different between subject conditions.

For detailed documentation and samples see:

http://www.empirisoft.com/support/showthread.php?p=826

Sending and Receiving Serial Data

Some external devices can emulate keyboard input so that input will be instantly recognized as key presses from a standard keyboard--they will just arrive much faster. For example, our DirectINTM Precision Button Boxes and Keyboards are all USB ready and send signals within 1ms that look like regular keystrokes to your computer. Some older devices though still communicate via the serial port. This section describes how you can get MediaLab to recognize and record serial port data.

Receiving a Single Response via Serial Port

As an alternative to keyboard input, you can receive input from external devices such as response boxes via the serial port. To do so you just need to create a file called *comport.txt* in your experiment folder and copy the following text into it:

```
comPort, baudRate, parity, dataBit, stopBit
1,19200,n,8,1
input code (1-255), response value (1-12)
1,1
2,2
3,3
4,4
5,5
6,6
```

You should not modify the first or third line. The second line represents the values of the variables listed on the first line. You can change any of these as necessary. The remaining lines tell MediaLab how to map incoming signals to response keys. On each line you can list a pair where the first value is the signal sent to the serial port (i.e., 1-255) and the second value is the response it should map on to (i.e., 1 to 12). In the example above, if MediaLab detects a "1" coming through the serial port, it will react as though the 1 or F1 key had just been pressed.

Receiving Streams of Serial Data

MediaLab can also accept and log *streams* of serial data from an external source. MediaLab will write the exact ASCII data to an external file named *comdata.txt.* To indicate to MediaLab that the serial data should be recorded, your serial device should be programmed to send out data packets that are surrounded in {} brackets as follows:

```
{
Hello World
This is data for MediaLab
}
```

The *comdata.txt* file will contain the data

```
Hello World
This is data for MediaLab
```

This function requires that you create a file called *comdata.txt* and place it in your experiment folder. If MediaLab sees this file, it will automatically watch for and save any serial data that is surrounded in { } brackets. In order for MediaLab to use the proper serial port settings, you should also include a comport.txt file as described above. If you do not want any key mappings, it could simply look like this:

```
comPort, baudRate, parity, dataBit, stopBit
1,19200,n,8,1
input code (1-255), response value (1-12)
0,0
```

Sending Streams of Serial Data

If you are using a *comport.txt* file, you can also send a string of data to the serial port by adding the string as a parameter value for any given questionnaire item. The string should be enclosed within braces-{}- and angular brackets-<>- like this:

```
<{112340}> or <{RED}>
```

When included in the parameters for a 5pt scale response, for example, your parameters field might look like this:

```
(p5,<{SCALE1}>)
```

In this example, so long as you have a valid *comport.txt* file in your experiment folder (see above), then MediaLab would send "SCALE1" to the appropriate port at the onset of this scale response item. Again, if you do not want any key mappings, *comport.txt* could simply look like this:

```
comPort, baudRate, parity, dataBit, stopBit
1,19200,n,8,1
input code (1-255), response value (1-12)
0,0
```

See Also:

Receiving Responses Via TTL Signals 62
Receiving Response Via Serial Port 55

11.4 Trouble Shooting Tips

Answers to many common trouble shooting questions can be found at our support site. If you can not find the answer to your question there, you may register a username and password and post your question directly on the support site. This is often the fastest way to get an answer from us or maybe even one of our other helpful users. If for some reason, you can not get the answer you need via the support site, then contact us directly through email at support@empirisoft.com.

No matter which route you take, there are two basic types of trouble you might experience while trying to use MediaLab. The first is trouble associated with a particular input file and the second is trouble associated with the program in general. Here are some questions and we're likely to ask and recommendations for what to do depending on your situation.

Questionnaire or Experiment File Trouble

If you're having trouble with a particular file you're working on, but otherwise MediaLab is running fine (e.g., samples work ok), then:

- Post your questions on the forums at the Empirisoft website or send us an email with your questions and a copy of your problem file with as few items in it as possible to illustrate your question. Ideally, try to send files that are directly relevant to your problem or question; this will help us see as quickly as possible what's going on. If we can quickly see and replicate the issue here, then we can usually solve the problem much faster. If you need to send a large file or set of files, it's best to use a compressed .zip folder.

System or Software Trouble

If it's an apparent bug that you've found, or some unusual behavior, try to tell us the following in your post or email:

- Does the same thing happen on multiple machines, or just one? Problems that replicate are usually much easier to fix!
- Is it seemingly random or predictable? What do you have to do to make it happen?
- Does the same thing happen with any of the files that come with MediaLab in the *Samples* folder or is it specific to a single file that you've created (see above)?
- What version of MediaLab (e.g., ver20xx.x.xxx) are you using? You can find this by clicking "About" on the Help menu. What version of Windows are you using? Have you tried making sure that no other programs are running at the same time that might be causing some kind of interference?

Note that the samples that come with MediaLab should always run fine. If they don't, you may have a system problem. In this case, the first thing to try is usually making sure you have the latest version of MediaLab.

One thing we can do is look over your system diagnostics and see if there are any obvious problems that could be corrected. If you want to try that, here is how you do it: Type "dxdiag" from the "Run" command on the Windows "Start" menu. You will get a complete diagnostic description of your system. If you then click "Save All Information" and send the resulting file to us, we'll take a look at it and see if we can see anything problematic. You can also check for some obvious problems here yourself--the display tab in particular may indicate you could benefit from a driver update or other action.

A

Absolute vs Relative Paths, 104
Advanced Features, 99
  Calculated Scores and Variables 101
  Complex Skips 102
  Session Report 102
Alternate Data Folder, 9, 83
Animation (see Movie), 47
Autostart Files, 87

B

BackGround,
  .exp 27
  .que 39
BackSound,
  .exp 28
  .que 40
BackVideo,
  .exp 28
  .que 41
ByQuestionnaire Data, 70
ByVariablename Data, 70

C

CD-ROM Experiments, 9
Clear Data, 75, 84
Codes, License, 4
Colours, 83
Command Line, c Parameter, 31, 57
ComPort Data, 107
Copying and Pasting in Editor, 17
CSV Data Files, 70, 73
Custom Items, 43
  Creating 88
  Data 76

D

Data,
  Access MDB file 75
  ByQuestionnaire 70
  ByVariablename 70
  Clear Data 75, 84
  CSV 70, 73
  Custom Items 76
  Determining Random Order 74
  Essay Responses 74
  Excel 73
  FAQ 72
  Merging Files 73
  Missing Values 75
  Recoding 74
  SAS 73
  SAV 70
  SPSS 70, 73
  Timestamps 74
  TXT 70
  Variable Names 76
  Vertical Format 75
  Write Data as Text 69
  Write Reaction Time 69
  Write to Data File 68
DirectRT, 43
  Using MediaLab Responses 106
DirectX, 109
Disable Clear Data, 84
Disable System Key Sequences, 83
Double-Click to close MediaLab, 83
Duration, d Parameter, 31, 57
Duration, d Parameter, .que, 60
dxdiag, DirectX, 109

E

Essay, 44
  Data 74
  Parameters 57
Excel, 70, 73
Executable, 44
  Parameters, .exp 31
  Parameters, .que 57
Experiment Editor, 11, 17
  Edit Menu 14
  Editing Tips 17
  File Menu 12
  Options 15
  Printing 16
  Windows 15
Experiment Files, 6, 20
  BackGround 27
  BackSound 28
  BackVideo 28
  Editing 20

© 1995-2014 Empirisoft Corporation

Experiment Files, 6, 20
Editing Tips 17
File 26
Overview Window 21
Parameters 31
Printing 16
Quitting Early 10
Randomize Between Groups, RBG 30
Randomize Within Groups, RWG 29
Running 9

F

Features,
Overview 1
Feedback, Modifying, 86
File Paths, 9, 104
Filename, .exp, 26
Filename, .que, 41, 53
Fill in the Blank, 44
Parameters 57
Fonts,
Fonts and Colours 83
Foreign 85

G

GoBack, 42

H

Height, w Parameter, .que, 63
Help, 109
Hide MediaLab Rats, 83
HTML,
Custom Items 43
Experiment File 26
Parameters, .exp 31
Parameters, .que 57
Question Labels 55
Questionnaire Item 45

I

Image,
BackGround, .exp 27
BackGround, .que 39
Experiment File 26
Parameters, .exp 31
Parameters, .que 57
Questionnaire Item 46
Inquisit, 47
Installation, 4
Instructions, 47
Parameters 57
IP Licensing, 4
Item Types, 43
Item Wording, 53

K

Key Concepts, MediaLab, 6
Keys to Skip Items, 10

L

Labels, 55
License Codes, 4
Licensing, IP, 4
Load Other Preferences File, 84

M

Machine Code, 84
Mask, m Parameter, 31, 57
MediaLab,
Data 70
Experiment Editor 11
Experiment Files 20
Main Program Window 8
Overview 1, 6
Preferences 82
Questionnaire Files 34
Trouble Shooting 109
MediaLab Sample Experiments, 76
Merging Data Files, 73
Message Text, Modifying, 86
Missing Values, 75
Modify Message Text, 86
Movie,
BackVideo, .exp 28
Backvideo, .que 41
Experiment File 26
Parameters, .exp 31
Parameters, .que 57
Questionnaire Item 47
Multiple Choice (see Scale Response), 50
Multiple Response, 47

© 1995-2014 Empirisoft Corporation

Multiple Response Item,
Parameters 57
Mute MediaLab Sounds, 84

N

Name, 56
Navigation, 10
GoBack 42
Secret Keys 10
Network Drives, 9
Number of Thoughts, n, 61
Number of Thoughts, n Parameter, 31, 57

O

Omit Conditions, 57
On-Line Ratings, 48
Parameters 57
Onset, o Parameter, 31, 57
Open Ended Responses,
Essay 44
Fill in the Blank 44
Options, Experiment Editor, 15
Options, MediaLab, 82
Overview Window,
Experiment Editor 21
Questionnairre Editor 35
Overview, MediaLab, 6

P

Parameters, .exp, 31
Parameters, .que, 57
Command Line Arguments, c 59
Duration, d 60
Height, h 63
Key, k 60
Left, l 60
Mask, m 61
Number of Points, p 61
Number of Thoughts, n 61
Onset of Wording, o 61
Password, x 61
Quickstyle, $ 59
Range, r 62
Spacing, s 62
Target Question, q 62
TimeStamp, @ 59
Top, t 60
TTL Signal 62
Width, w 63
Password to End Option, 84
Password, x Parameter, 31, 57
Path Names, 9, 104
Points, p Parameter, 31, 57
Position of Items, 64
Power Point Show,
Experiment File 26
Parameters, .exp 31
Questionnaire Item 48
Preferences, Experiment Editor, 15
Preferences, MediaLab, 82
Autostart Files 87
Disable Clear Data 84
Disable System Key Sequences 83
Double-Click to close MediaLab 83
Fonts and Colours 83
Foreign Fonts 85
Hide MediaLab Rats 83
Load Other Preferences File 84
Modify Message Text 86
Mute MediaLab Sounds 84
On-the-fly Editing of Word Documents 83
Password to End Option 84
QuickStyles 84
Saving Preferences 84
Supporting Applications 84
Unique Machine Code 84
Use Alternate Data File 83
Printing, 16
Probe Task, 106

Q

Question Wording, 53
Question, q Parameter, 31, 57
Questionnaire Editor, 34
Questionnaire Files, 6, 26, 34, 53
BackGround 39
BackSound 40
BackVideo 41
Custom 43
DirectRT 43
Editing 34
Editing Tips 17
Essay 44

© 1995-2014 Empirisoft Corporation

Questionnaire Files, 6, 26, 34, 53
Executables 44
Filename 41, 53
Fill in the Blank 44
GoBack 42
HTML 45
Image File 46
Inquisit 47
Instructions 47
Item Types 43
Item Wording 53
Labels 55
Movie 47
Multiple Response 47
Name 56
Omit Conditions 57
On-Line Ratings 48
Overview Window 35
Parameters 57
Position of Items 64
Power Point Show 48
Present n of x items 66
Printing 16
Quitting Early 10
Randomize Between Groups, RBG 65
Randomize Within Groups, RWG 64
Ranking 49
Recall List 49
Running 9
Scale Response 50
Skip To 66
Sound File 50
Text Labels 67
Thought Listing 51
Thought Rating 51
Voice Response 51
WebTracker 52
Word Document 52
Write Data as Text 69
Write Reaction Time 69
Write to Data File 68
Quickstyle, $ Parameter, 57
QuickStyles, 84
Quitting Early, 10

R

Randomization,
Present n of x items 66
RBG, .exp 30
RBG, .que 65
RWG, Experiment Files 29
RWG, .que 64
Tracking Presentation Order 74
Randomize Between Groups, .exp, 30
Randomize Between Groups, .que, 65
Randomize Within Groups, .exp, 29
Randomize Within Groups, .que, 64
Range, r Parameter, 31, 57, 62
Ranking, 49
Parameters 57
Recall List, 49
Parameters 57
Recoding Data, 74
Reinstalling, 4
Relative vs Absolute Paths, 104
Response Times,
Variable Names 56
Running a Session, 9

S

Sample Experiments, 76
SAS, 73
SAV Data Files, 70
Saving Preferences, 84
Scale Response, 50
Parameters 57
Screen Resolution, 4
Secondary Task, 106
Secret Keys, 10
Serial Data, 107
Setup, MediaLab, 4
Skip To, 66
Sound,
BackSound, .exp 28
BackSound, .que 40
Experiment File 26
Parameters, .exp 31
Parameters, .que 57
Questionnaire Files 50
Sound Effects, MediaLab, 84
Spacing, s Parameter, 31, 57, 62
Speaker Volume, 106
Speaker,Volume,Sound, 107
Spell Checking, 20

© 1995-2014 Empirisoft Corporation

SPSS, 73
  Syntax and Data 70
  Variable Names 56
Support, 109
Supporting Applications, 84
System Requirements, 4
System Settings, 4

T

Target Question, q Parameter, 62
Task bar, 4
Text Labels, 67
Thought Listing, 51
Thought Listing,
  Parameters 57
Thought Rating,
  Parameters 57
Timestamp, @ Parameter, 57, 74
Trouble Shooting, 109
TTL Signal Parameter, 62
TXT Data Files, 70

U

Unique Machine Code, 84
Use Alternate Data File, 83

V

Variable Names, 56, 76
Video (see Movie), 47
Voice Response, 51
  Parameters 57
Volume, Speaker, 106

W

WebTracker, 52
Width, w Parameter, .que, 63
Word Document,
  Experiment File 26
  On-the-fly Editing 83
  Parameters, .exp 31
  Parameters, .que 57
  Questionnaire Item 52
Write Data as Text, 69
Write Reaction Time, 69

Z

Zip Files, 76

© 1995-2014 Empirisoft Corporation