

# Table of Contents

<a href="#">Abstract</a>	The Sustained Operations (SUSOPS) experiments are designed to test how various aspects of cognitive ability are affected by stresses such as sleep deprivation.
<a href="#">Overview</a>	The Sustained Operations (SUSOPS) experiments are designed to test how various aspects of cognitive ability are affected by stresses such as sleep deprivation.
<a href="#">Schedule Editor</a>	The Schedule Editor is used to create and modify lists of tasks. Each list is known as a schedule. The schedule also says how long each task should take.
<a href="#">Dispatcher</a>	The ads a schedule file and runs the tasks listed in it one at a time. It also keeps a list of how many subjects have run the schedule, and how many sessions they have each run.
<a href="#">Environment</a>	
<a href="#">Task Battery</a>	
<a href="#">Questionnaire Editor</a>	The Questionnaire Editor allows the experimenter to create and edit multiple-choice and short answer questionnaires. This is done with a visual approach, so the editing session looks much like the final product.
<a href="#">Distributing PenSUSOPS System</a>	Typically, a Scientist creates and tests a schedule on their own computer but runs it on one or more others that will be used by subjects. Further, a Scientist might wish to send an experiment to a colleague in another laboratory.
<a href="#">Maintenance and Support</a>	PenSUSOPS software and documentation was developed by NTT SYSTEMS INC. under contract to the Canadian Department of National Defence. It is copyright Her Majesty the Queen in Right of Canada as represented by the Minister of National Defence.
<a href="#">About</a>	About PENSUSOPS

Created Using docVB - Licensed to Marc Grushcow, NTT Systems Inc.

# Abstract

## Description

The Sustained Operations (SUSOPS) experiments are designed to test how various aspects of cognitive ability are affected by stresses such as sleep deprivation.

In the past the experiments were carried out using a VAX mainframe computer, which meant that the experiments were confined to the lab. The PenSusops project moved the experimental tasks to a portable computer with a pen-based interface. This means that PenSusops trials can be carried out in the field.

In addition a new program was provided to create experimental schedules and organise the resulting data. All the tasks produce data with records of the same format.

Some new tasks have been added. One of them, a questionnaire, also has a flexible editor to create new questions.

PenSUSOPS has been used by different organizations in a number of environments. A distribution facility has been added to the system that can be used to automatically package the software and experiments on one computer and set them up again on another. This automated approach ensures that the distributed system contains all of the components needed to author and run experiments.

# Overview

## Description

The Sustained Operations (SUSOPS) experiments are designed to test how various aspects of cognitive ability are affected by stresses such as sleep deprivation.

The basic unit of organization is the schedule. You (as the experimenter) create a list of timed tasks to be performed in a single sitting. This could be something like 3 minutes of a logic task, followed by 5 minutes of addition, 1 minute of rest, and then 3 more minutes of logic. To do this you use a program called the Schedule Editor.

Each schedule is kept in its own directory (named after the schedule). This directory also holds the result data for the tasks on that schedule. For historical reasons the schedule name is also known as the experiment name.

The schedule is run by a program known as the Dispatcher. It reads the schedule and starts each task in turn. Each task is told how many seconds it should run. When the task terminates the dispatcher will start a new task.

When the experiment is over you will find the data all collected together in the directory. Each subject in the experiment produces one data file per task.

# Schedule Editor

## Description

The Schedule Editor is used to create and modify lists of tasks. Each list is known as a schedule. The schedule also says how long each task should take.

## Topics

### [Main Screen](#)

Here is a picture of the Schedule Editor with a schedule. The list of tasks takes up the bulk of the window (the grey area). (The "end" line shows the total duration of the schedule.)

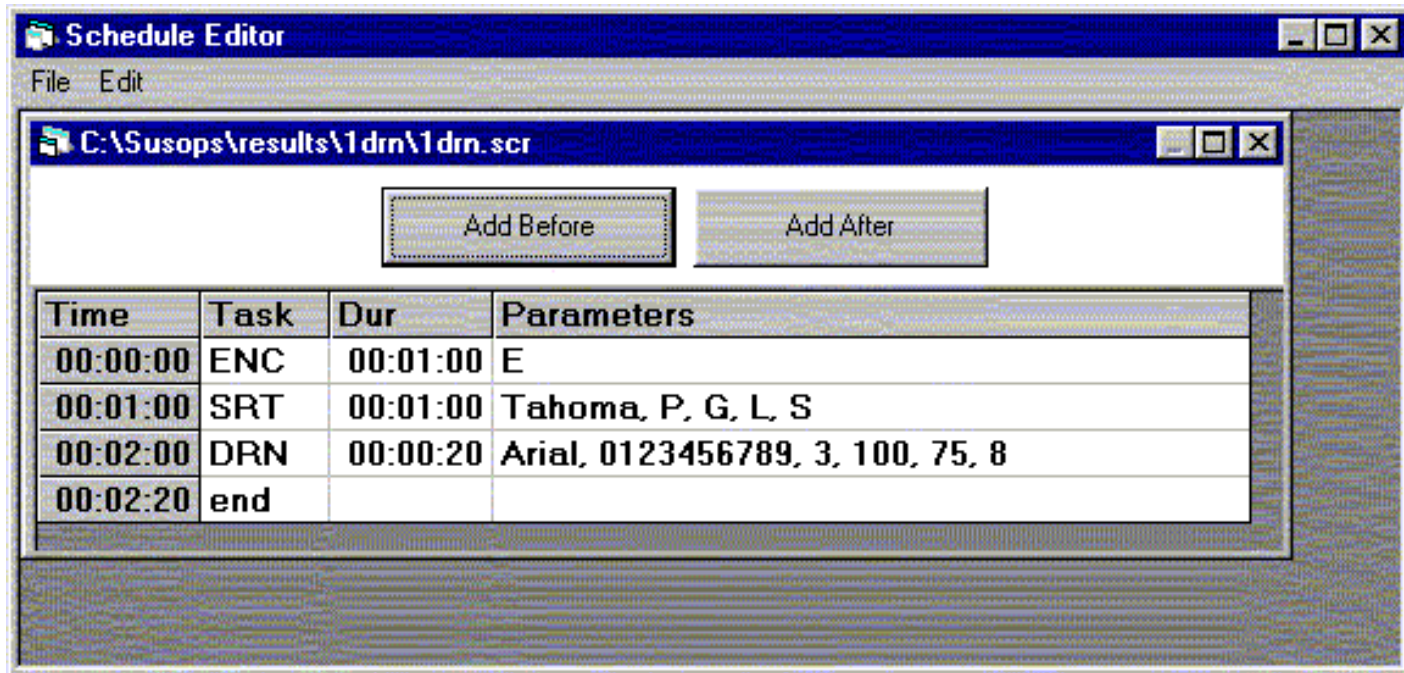
### [Menus](#)

The two menus are File and Edit. The File menu allows you to read and save schedules. The Edit window lets you modify the current schedule.

## Main Screen

### Description

Here is a picture of the Schedule Editor with a schedule. The list of tasks takes up the bulk of the window (the grey area). (The "end" line shows the total duration of the schedule.)



We can go over the columns of the task listing.

#### Time

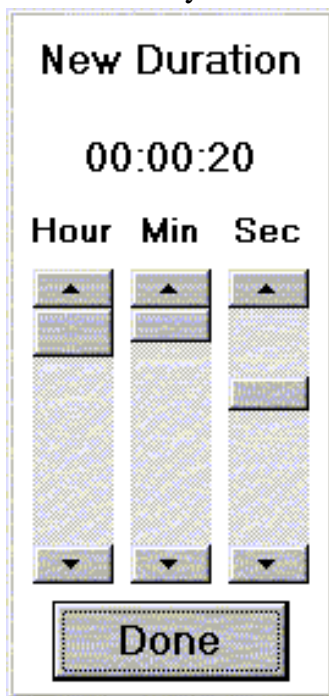
The first column lists the cumulative time of the tasks in the schedule. The total time for one session is listed at the end. As you can see it is 2 minutes and 20 seconds. Proper display of Time and Duration depends on your Windows time format being set to 24 hour mode. This is done through the "international" icon on the Windows Control Panel.

Task	This is the task to be performed. Each task has a three character identifier. The tasks are shown below.
Dur	This is the length of time that the subject must perform the task. ("Dur" is short for Duration.)
Parameters	These are bits of information specific to the task. They include names of files, timeouts, etc. There are separate parameter editors for each task, and they are described later.

You can do several things by clicking on various parts of the task list.

## ○ Duration

If you click on the Duration column then you get a new window that lets you change the amount of time for a task. You have three sliders to change hours, minutes, and seconds. Click on the Done button when you have set the time you want.



## ○ Parameters

If you click on the Parameters column then you get a new window that depends on the task. These windows describe the parameters, and let you change them to new values. The parameters are described later with each individual task.

## ○ Selecting Tasks

If you click on the Task column (or the Time column) you select a line. If you Shift-Click on another line you will select the range of lines between your click and shift-click. You can also set a range of lines by Dragging the pen across them.

- The selected line(s) can be Cut or Copied, or used by Add After or Add Before, all described later.

## Add Before, Add After

You add new tasks to the schedule by clicking on the Add Before or Add After buttons. Both call up a list of tasks for you to choose from. (There is also a "forget it" choice to cancel the list.) After you pick a task, a Parameter window will open. Once you have chosen new parameter values the new task will be added to the schedule.

If you clicked Add After then the line will be added after the last selected task. If you clicked Add Before then the line will be added before the first selected task.

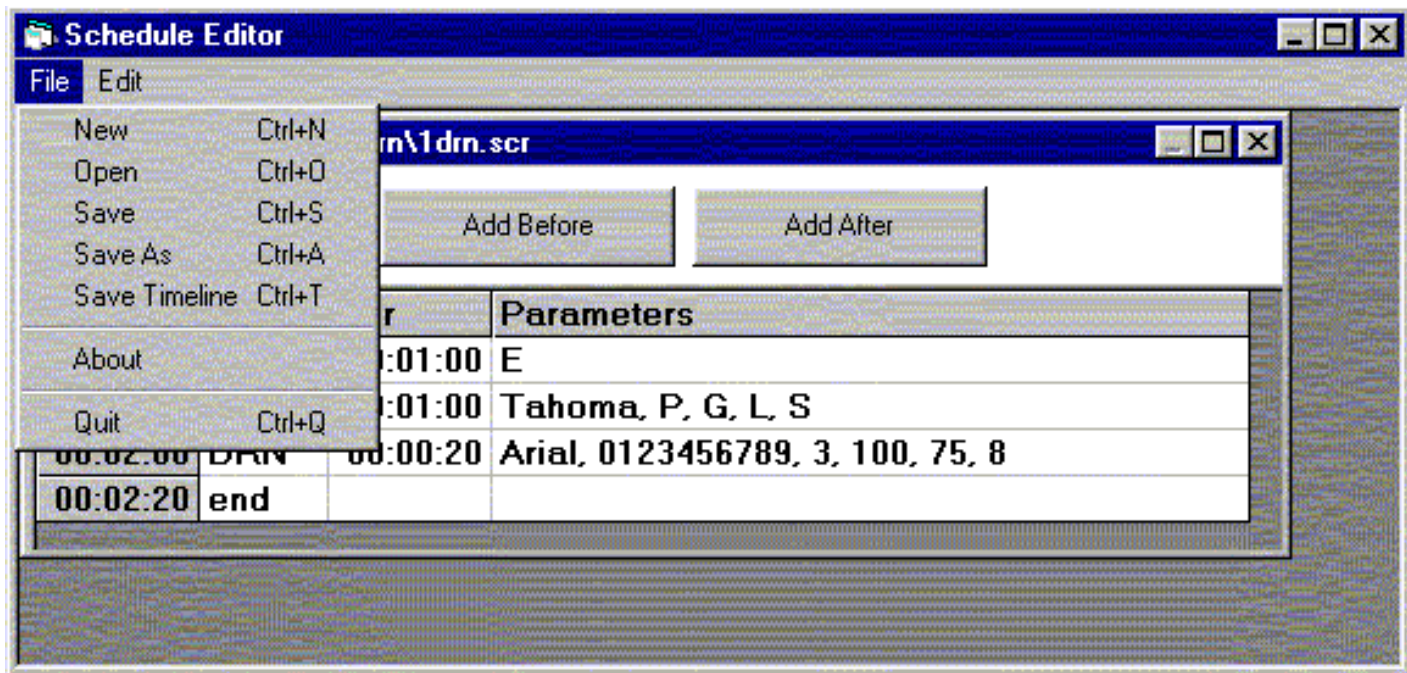


## Menus

### Description

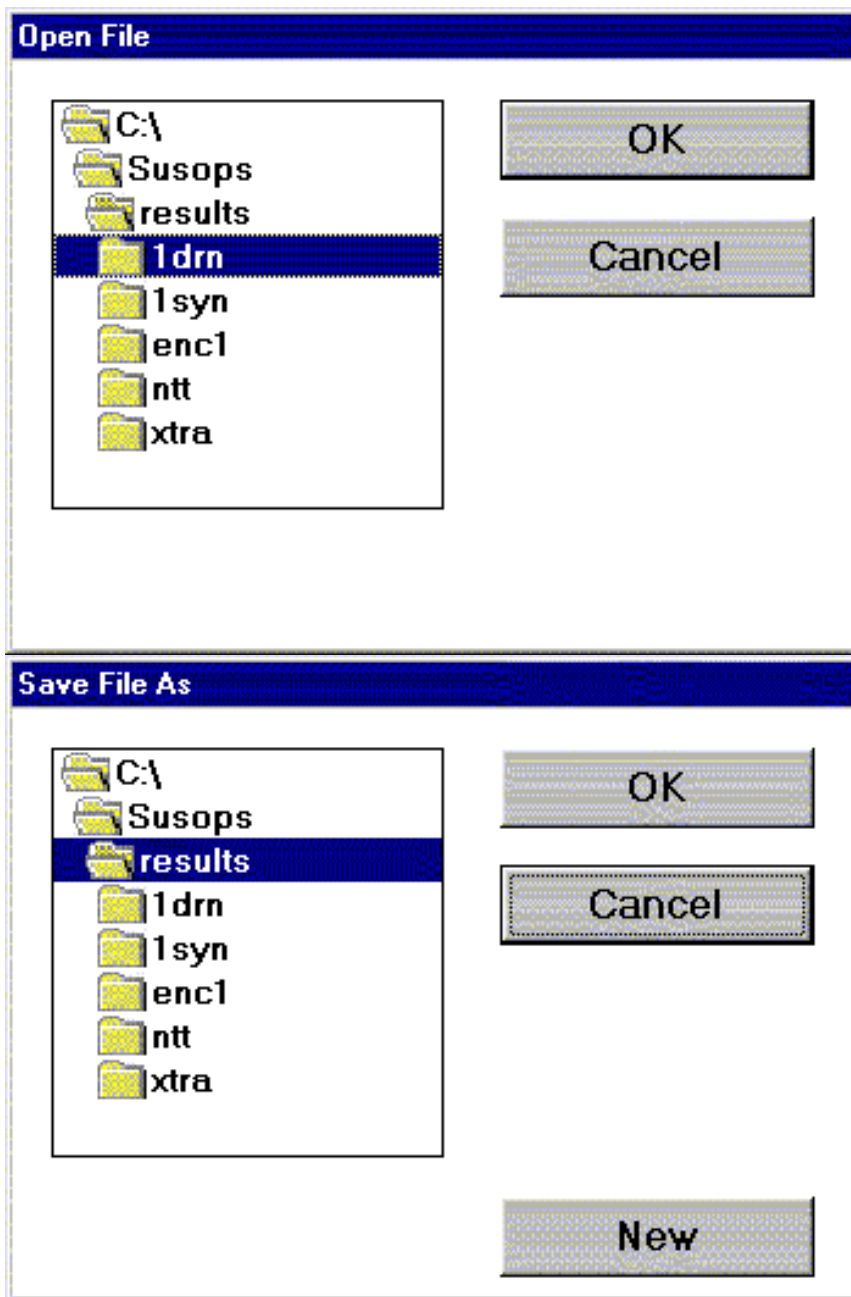
- The two menus are File and Edit. The File menu allows you to read and save schedules. The Edit window lets you modify the current schedule.

### File Menu

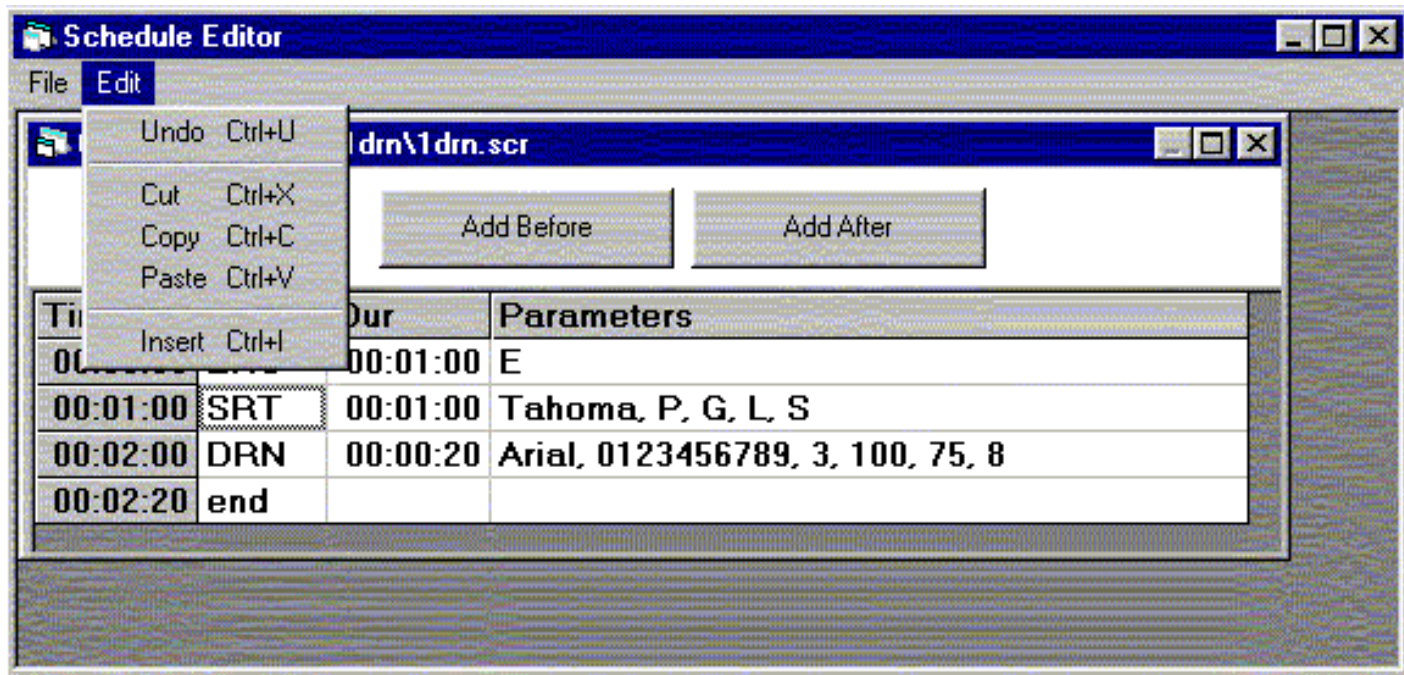


New	Starts a new schedule.
Open	Opens an existing schedule.
Save	Saves the schedule. If it is a new schedule, then it is doing Save As.
Save As	Saves a new schedule. The user is required to give a filename and schedule will be saved in the folder created at the same time using the same name. Refer to the <a href="#">Results Directory</a> for more information.
Save Timeline	Saves the schedule as it is. The user is required to provide a filename.
About	Displays general information about the schedule editor.
Quit	Closes the schedule. The program will prompt for save if any of the opened schedules have been modified or new schedule has been created.

Note: The file requestor for Open, Save and Save As is non-standard. It shows the directories that currently contain schedules. If you wish to choose one of them then click on the directory, and then click on the OK button. If you want to change your mind then click the Cancel button. If you want to create a new directory then click on the New button. You will then be asked for a new name (maximum 5 characters). The windows can be seen below:



- **Edit Menu**



- Undo           Undos the last change. If multiple changes have been made, the program only allows the schedule recover the last change. Each opened schedule has its own undo.
- Cut            Copies all the selected lines to a cut-and-paste buffer, and then deletes them from the schedule.
- Copy           Copies all the selected lines to the cut-and-paste buffer, without changing the current schedule.
- Paste          Copies all the lines from the cut-and-paste buffer and inserts them in the schedule before the selected line.
- Insert         Inserts a schedule after the current selected line. The user will be asked to select the script folder and the script will be inserted.

Note: Cut, Copy, and Paste shares one cut-and-paste buffer. The program allows to cut (or copy) and paste to either the same schedule or different opened schedules.

## Results Directory

All the schedules are kept in a directory named "Results". Each schedule has its own directory named after it, and is stored in a file also named after it ending with ".scr".

For example if a schedule is named "xyz" then the schedule is kept in the file named ".../Results/xyz/xyz.scr".

## How to Run It

The PenSUSOPS Setup program creates an icon for the Schedule Editor in the group that it creates on your desktop. It is set up so that your scripts are automatically placed in the expected directory. Run the Schedule Editor by double clicking on its icon.

# Dispatcher

## Description

The Dispatcher reads a schedule file and runs the tasks listed in it one at a time. It also keeps a list of how many subjects have run the schedule, and how many sessions they have each run.

The tasks that are currently run by the Dispatcher all have a similar set of characteristics. They all accept a common set of parameters, including a timeout for the task (after which it dies) and a filename to store the results in.

## Topics

### [How to Run It](#)

The PenSUSOPS Setup program creates an icon for the Dispatcher in the group that it creates on your desktop. It will automatically find the directory that contains your schedules so that you can select the one you want. Run the Dispatcher by double clicking on its icon. When it starts you will see this window:

### [Schedule File](#)

The schedule file is described in the Schedule Editor documentation above. It lists the tasks to be run, the length of time each should run, and any task-specific parameters.

### [Sessions File](#)

The sessions file keeps track of which subjects have run this schedule, and how many times they have run it. In the normal course of things you will not need to edit it. It is kept up-to-date by the Dispatcher.

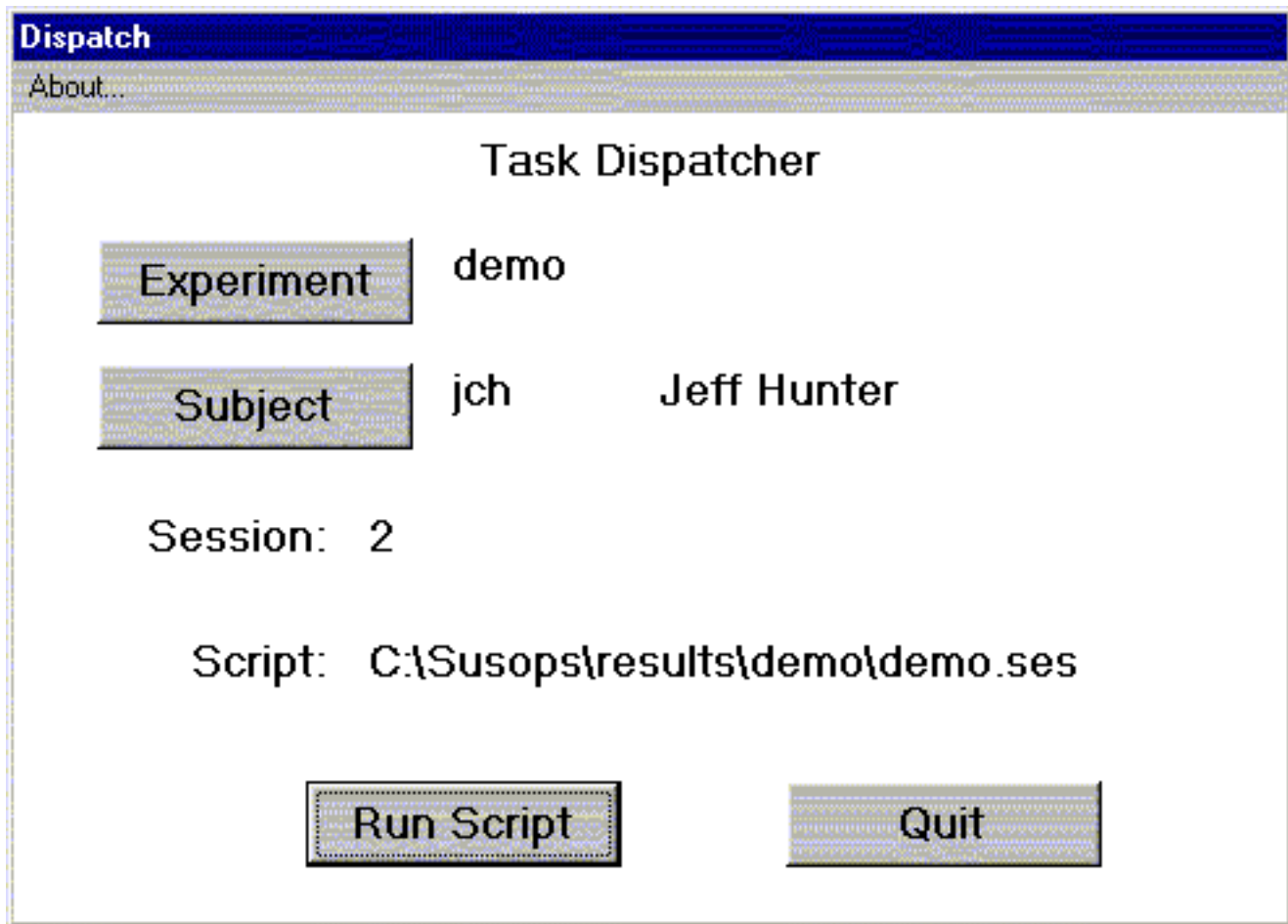
### [Rolling Back A Session](#)

A run of a schedule may die part way through if there is some problem with the computer or the subject. It is assumed that you will merely start another session, and later discard the partial data.

## How To Run It

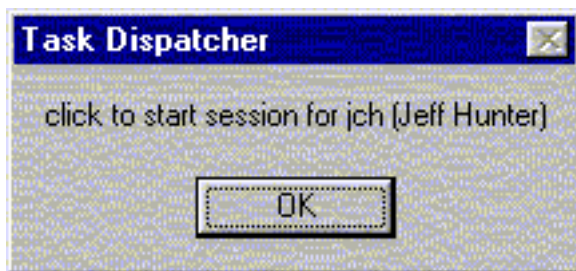
### Description

The PenSUSOPS Setup program creates an icon for the Dispatcher in the group that it creates on your desktop. It will automatically find the directory that contains your schedules so that you can select the one you want. Run the Dispatcher by double clicking on its icon. When it starts you will see this window:



Click on the Experiment button and choose a schedule to run. Then click on the Subject button to pick one of the existing subjects or to add a new one.

Then Click on the Run Script button to run the schedule. You will see the screen go white, and then you will see a small window asking you to click to start.



You can now give the computer to the subject and let them run the tasks. When the tasks are all complete another window will appear saying that all tasks are done.



# Schedule File

## Description

The schedule file is described in the Schedule Editor documentation above. It lists the tasks to be run, the length of time each should run, and any task-specific parameters. Refer to [Schedule Editor](#) for more information.

# Sessions File

## Description

The sessions file keeps track of which subjects have run this schedule, and how many times they have run it. In the normal course of things you will not need to edit it. It is kept up-to-date by the Dispatcher.

The session file is kept in the schedule directory, and has a name ending in ".ses".

For example if a schedule is named "xyz" then the session file is named ".../Results/xyz/xyz.ses".

There is one line per subject. The line contains the subject identifier (a three letter or number code for the subject), the number of times the subject has run the schedule (the session count), and an optional "real name" for the subject.

Here is a sample session file.

```
jch      2      Jeff Hunter
```

# Rolling Back A Session

## Description

A run of a schedule may die part way through if there is some problem with the computer or the subject. It is assumed that you will merely start another session, and later discard the partial data.

You may however wish to remove the partial session results, and re-run a session. There is no automated way to do this.

If you really wish to remove a session here are instructions. You can only do this immediately after the aborted session. Once the subject has completed another run its too late.

Pick any editor to edit the files with. The Notepad is handy, but will not work for really big data files.

Edit the session file. Find the line for the subject. Decrease the session number by one. (Be sure not to remove the TABs between the the columns of data.)

Look for data files with names ending in the subject identifier. (For example if the schedule is "xyz" and the subject ID is "123" then the data will be in files with names like "xyzLRT.123" and "xyzSTM.123".

Go to the end of each file and delete all the records with the session number that you are want to remove. The session number is the third field in each line.

# Environment

## Topics

- [Directory And File Structure](#)

All the files for PenSusops are collected together in one main directory. This should be the "current" directory when any of the PenSusops programs are run. (For example these programs were all tested in the "C:\pentest" directory.)

- [Standard Results File Format](#)

The current set of tasks save their data in a common format. (Except for the questionnaire, which uses the standard format only for a header line.)

## Directory And File Structure

### Description

All the files for PenSusops are collected together in one main directory. This should be the "current" directory when any of the PenSusops programs are run. (For example these programs were all tested in the "C:\pentest" directory.)

There are several subdirectories. The programs (Schedule Editor, Dispatcher, Questionnaire Editor, and all the tasks) are collected in a directory named "programs".

The text files for the Instruction Task are collected in a directory named "instruct".

The trivia files for the General Knowledge task are collected in a directory named "question".

The questionnaire files for the Questionnaire task are collected in a directory named "qst".

Finally the schedules and data from the experiments are collected in a directory named "results". Each schedule has its own subdirectory.

In each of these subdirectories there is a schedule file, a session file, and several data files. Each subject in the experiment produces one line in the session file, and one or more data files. (Each task creates a new data file for each subject. Thus the number of data files depends on the number of tasks in the schedule.)

For an example lets take a schedule named "day" that contains two tasks, the LRT (Logical Reasoning) and the SRT (Serial Reaction Time). After running two subjects ("ann" and "bob") the directory structure would look like:

```
instruct
  ... instruction files
programs
  ... executable programs
```

```

question
    ... trivia lists
qst
    ... questionnaires
results
    day
        day.scr
            - the schedule file
        day.ses
            - the session file

        dayLRT.ann
            - data files ...
        dayLRT.bob
        daySRT.ann
        daySRT.bob

```

# Standard Results File Format

## Description

The current set of tasks save their data in a common format. (Except for the questionnaire, which uses the standard format only for a header line.)

There is one line per trial. For example in the Logic task one record is produced for each question the subject answers. Enough information is placed on each line (in a standard record header) to identify when and why the data was produced. Task-specific information is placed at the end of each line. One field (the date-time) is only printed on the first record of a session. The other records have a null string in that field.

The standard record header layout is:

```

the name of the experiment (i.e. the schedule)
(up to 5 characters)

```

```

the identifier of the subject (up to 3 characters)

```

```

the session number (how many times this subject
has run in this experiment)

```

```

the task identifier (3 characters e.g. QST)

```

```

the block number (eg. how many questionnaires have
been presented so far this session)

```

the trial number (how many trials the subject has performed in the current block)

the date and time the task was started

the number of seconds that has elapsed from the start of the task of this trial

All fields are tab delimited to facilitate importing of results files into spreadsheets and other analysis programs. Missing data items are encoded as "." (period or dot).

# TASK

## Topics

- [ADD -- Addition](#)

The Addition task presents 8 numbers one at a time, which the subject must mentally add. The subject is asked to enter the sum, and then to indicate how confident they are that the answer is correct.

- [ASD -- Auditory Signal Detection](#)

A sequence of tones are presented to the subject. Tones are either long or short. The subject has to detect short tones and press a key or click a mouse button as quickly as possible.

- [CMP -- Line Comparison](#)

The Line Comparison task asks "Longer" or "Shorter" and then presents two lines. The subject indicates if the left or right line is longer (or shorter). The subject is then asked to rate their confidence from 50% to 100%.

- [DRN -- Detect Repeat Number](#)

The Detect Repeat Number task will present a sequence of strings on a monitor. Successive strings differ by one character. During each minute, there is an experimenter specified number of occasions when the same string is presented on successive trials. The subject has to detect these and press a key or click a mouse button as quickly as possible.

- [ENC -- Encoding and Decoding](#)

The subject is presented with map co-ordinates to code or decode. Coding and decoding trials alternate. Coordinates consist of a three digit Easting followed by a three digit Northing. Each is encoded by converting the first two digits into a letter and the third digit into a separate letter. The subject may input an answer by clicking on the keys or by pressing the corresponding keys on the keyboard. The mapping is done as follows:

- [GKN -- General Knowledge Quiz](#)

The subject is presented with a simple question and two possible answers. They then indicate how confident they are in their answer.

- [GK2 -- General Knowledge Task 2](#)

The GK2 General Knowledge Task 2 presents subjects with a series of problems in which they have to select one of two alternatives based on a specified criterion. In each problem, the criterion is displayed as a prompt in the center of the screen. An example of a prompt is "Which is larger?" (in reference to population). Following a delay, two alternatives are displayed in the lower left and lower right portion of the display. The subject makes a selection by depressing the corresponding button (ie. left or right).

- [GTP -- General Text Presentation](#)

The General Text Presentation Task (GTP) displays a series of prompts on the screen and records the subject's response to each. Prompts are formatted strings read from a file specified on the command line and can be one or more lines long. The subject can respond by mouse or keyboard as described below.

- [INN -- Instruction](#)

The instruction task displays some text to the subject for a given amount of time. This text is usually instructions on how to do the task which follows. The text comes from a file.

- [LRT -- Logical Reasoning](#)

The Logical reasoning task presents a series of problems concerning the relationship between two entities A and B. A proposition is displayed of the form entity relationship other-entity. Where relationship can be one of the following:

- [QST -- Questionnaire](#)

The Questionnaire Presenter is a straightforward program that presents a questionnaire and records the answers. It does not attempt to record when, or in what order questions were answered.

- [SRT -- Serial Reaction Time](#)

The Serial Reaction Time task displays a four key keypad with a different character or symbol on each key. One of the four graphics will be shown in a display area and the subject must click on the corresponding key as quickly as possible. Display characters are selected from the four keypad characters randomly with replacement. Since this allows for sequential repetition of a character, the display colours are reversed from presentation to presentation so that the subject can detect the onset of a new stimulus.

- [STM -- Short Term Memory](#)

A series of digit strings are presented to the subject along with a direction indicator. If the indicator is "FORWARD" the subject must enter the digits in the order they were presented. If it is "REVERSE", they must enter the digits in the reverse order. The digit string length and direction of recall vary according to a set of rules. The initial recall direction is FORWARD and the initial string length is four digits.

- [SUB -- Subtraction](#)

The subject is presented with an initial subtraction problem. If Toughness is "EASY", the subtrahend is in the range 500...999 and the minuend is in the range 5...9. If it is "HARD", the subtrahend is in the range 500...9,999 and the minuend is in the range 12...36, with 22, 30 and 33 being excluded. The subject enters their answer by clicking on a numeric keypad or pressing the corresponding keys on the keyboard. Each time the required number of digits is entered the display is cleared (thus the subject can not edit the last digit). The subject continues by repeatedly subtracting the fixed minuend from their most recent result. No further input is accepted after the specified duration has elapsed. The task terminates after the specified rest period expires. The rest period allows several SUB tasks to be run with short durations sequentially in a schedule.

- [SYN -- Synwork Launcher](#)

The Synwork Launcher will run the stand alone version of SYNWORK. The experimenter is responsible for specifying a command line, along with arguments required, to run SYNWORK. The SUSOPS duration control will override the duration argument of SYNWORK.

## **ADD -- Addition**

## ● Description

The Addition task presents 8 numbers one at a time, which the subject must mentally add. The subject is asked to enter the sum, and then to indicate how confident they are that the answer is correct.

## ● The detailed task events are as follows

A "Get Ready" prompt is displayed for Ready Time seconds.

A set of 8 addends are generated. Each is presented for Digit Time seconds with a Inter Time delay between digits.

A numeric keypad is presented for input. If the "ENTER" key is not pressed before Answer Time seconds, the message "ANSWER NOW PLEASE" is displayed for Extra Time seconds.

The subject may input by clicking on a key on the keypad or by pressing a corresponding key on the keyboard.

If "ENTER" is pressed or time expires, the confidence scale is displayed until the subject picks a confidence value or Quest Time elapses.

The whole process is repeated until task duration expires.

Eight addends are generated for each trial. The addends are selected from the following six sets:

```

1, 2
2, 3, 4, 5, 6
4, 5, 6, 7, 8
6, 7, 8, 9, 12
8, 9, 12, 13, 14
12, 13, 14, 15, 16

```

Addend sets are selected randomly without replacement for each set of six trials. Addends are selected randomly with replacement.

## ● Parameters

The parameters for the Addition task are time-outs and duration (in seconds) for various parts of the task.

Answer\_Time:

initial wait for subject to answer (1..60).

Extra\_Time:

wait extension for subject to answer (0..60).

Quest\_Time:

wait for subject to express confidence 1..60).

Ready\_Time:

time to display "Ready" prompt (0..60).

Digit\_Time:

how long to present each number (1..5).

Inter\_Time:

how long between presenting each number (1..5).

Out of bounds parameters are set to their closest limit.

## ● Output Data

The Addition task uses the standard data format for the record header. The extra data fields at the end of the record are:

Response\_Time:

Time between presentation of trial and subject response (When the subject hit the Enter button.) (in milliseconds)

Subject\_Answer:

the subject's sum

Expected\_Answer:

the real sum

Confidence:

The subject's confidence level. (0 = definitely wrong or didn't answer, 50 = unsure, 100 = definitely right)

Addends:

The eight numbers that were added.

Parameters:

The six parameters described above. (in milliseconds)

# ASD -- Auditory Signal Detection

## ● Description

A sequence of tones are presented to the subject. Tones are either long or short. The subject has to detect short tones and press a key or click a mouse button as quickly as possible.

## ● The detailed task events are as follows

Randomly determine whether a short tone or a long tone should be presented.

Present the tone. If it is a short tone, the subject has to press a key or click a mouse button as quickly as possible. A score is recorded.

Silent for the rest of the period.

Repeat the whole process until the task duration expires.

## ● **Parameter**

The parameters for the Auditory Signal Detection are:

Frequency:

Frequency of the tone.

Long\_Tone\_Length:

Long tone's duration in seconds (eg., 0.25)

Short\_Tone\_Length:

Short tone's duration in seconds

Prob.\_of\_Short\_Tone:

Probability of generating short tone.

Period\_Length:

Length of a tone period, including silent period in seconds.

## ● **NOTE**

The minimum duration that can be reliably produced is 0.2 seconds. The precision is approximately 0.050 seconds. That is, durations of 0.3, 0.315, and 0.33 will all play for 0.300 seconds. Specifying 0.2 and 0.25 should reliably produce tones of 0.2 and 0.25 respectively.

## ● **Output Data**

The Auditory Signal Detection task uses the standard data format for the record header. The extra data fields at the end of the record are:

Response\_Time:

Response time of the subject.

Score:

Score of the response. There are four types of score: "C" for correct detection, "M" for missed detection, "F" for false detection and "R" for rejected detection.

# **CMP -- Line Comparison**

## ● **Description**

The Line Comparison task asks "Longer" or "Shorter" and then presents two lines. The subject indicates if the left or right line is longer (or shorter). The subject is then asked to rate their confidence from 50% to 100%.

There are two versions of this task. One is set up for pen input. The subject clicks on screen buttons labeled "R" and "L". The other is set up for mouse input. The subject chooses right or left by clicking the corresponding left or right mouse button. (This second version is named CM2.)

## ● Parameter

The parameters for the Comparison tasks are:

Maximum\_Trials:

The maximum number of trials to be presented.

Short\_Line-Length\_Pairs:

One to four pairs of line lengths.

Long\_Line-Length\_Pairs:

One to four pairs of line lengths.

Each pair of lines can be used to generate four problems by assigning the short line to the left or right side of the display and by asking "Longer" or "Shorter" for each configuration.

A problem subset is constructed by selecting one short pair and one long pair of lines and presenting all eight possible problems (four for each line pair as described above). The ordering of problems is random.

A problem set consists of all possible subsets, that is, all combinations of short pairs and long pairs. Problem subsets are selected at random without replacement.

A problem set can generate  $\text{numberOfShortPairs} * \text{numberOfLongPairs} * 8$  distinct trials. Trials are generated until the task duration or Maximum Trials is attained. If the entire problem set has been presented and time and trials remain, the process is repeated with a regenerated problem set.

## ● Output Data

The Comparison task uses the standard data format for the record header. The extra data fields at the end of the record are:

Response\_Time:

Time between display of lines and subject response (When the subject hit "L" or "R" button.) (in milliseconds)

Left\_Line\_Length:

Length of line displayed to left of median (in pixels)

Right\_Line\_Length:

Length of line displayed to right of median (in pixels)

Longer\_or\_Shorter:

0 if prompt was "Shorter", 1 if prompt was "Longer"

Subject\_Answer:

0 if subject clicked the left button, 1 for right

Correct:

0 if subject gave the wrong answer, 1 if subject was right

Confidence:

50 to 100 % confidence level

## DRN Detect Repeat Number

### ● Description

The Detect Repeat Number task will present a sequence of strings on a monitor. Successive strings differ by one character. During each minute, there is an experimenter specified number of occasions when the same string is presented on successive trials. The subject has to detect these and press a key or click a mouse button as quickly as possible.

### ● The detailed task events are as follows

Step\_one:

The task determines which of the presentations in the next minute will be repeated.

Step\_two:

An initial set of presentation strings is randomly generated from the specified character set with replacement.

Step\_three:

Display the set of string. If it is the same as the previous one, the subject should press a key or click a mouse button as quickly as possible. A score will be recorded.

Step\_four:

Generate a new presentation string. If it is to be a repeat, it will be identical to the previous string; otherwise, one of the characters in the current string is selected at random and replaced by a different character randomly selected from the character set.

Step\_five:

Display a blank screen.

Step\_six:

Repeat Step three.

Step\_seven:

When at the start of a new minute, repeat Step one.

### ● Parameters

The parameters for the DRN are:

Character\_Set:

A set of characters that is used to generate strings.

Font:

The font that will be used to display the strings.

Number\_of\_Characters:

Number of characters in each presentation string.

Cycle\_per\_Minute:

Number of cycles in a minute.

Percent\_on\_time:

Percent of the time in each cycle when the string is visible.

Repeats\_per\_Minute:

Number of repeats within a minute.

## ● Output Data

DRN uses the standard data format for the record header. The extra data fields at the end of the record are:

Minute\_in\_Trial:

Minute number within the task in which the string was presented.

Presentation\_in\_Trial:

Trial number.

Response\_Time:

The subject's response time.

Score:

Score indicates whether or not the subject properly classified the event. There are three types of score: "C" for correct detection, "M" for missed detection and "F" for false detection.

Current\_String:

The string that was used in the presentation.

Previous\_String:

The string that was used in the previous presentation.

# ENC -- Encoding and Decoding

## ● Description

The subject is presented with map co-ordinates to code or decode. Coding and decoding trials alternate. Coordinates consist of a three digit Easting followed by a three digit Northing. Each is encoded by converting the first two digits into a letter and the third digit into a separate letter. The subject may input an answer by clicking on the keys or by pressing the corresponding keys on the keyboard. The mapping is done as follows:

## ● Eastings

A series of 26 digit pairs are displayed in a row. The values go from 87 to 99 and from 00 to 12. For each trial the letters A--Z are randomly distributed over the digit pairs. To encode 123, find the letter over the 12 (X for example), and either the letter over 93 (say, Q) or the letter over 03 (say, V). Thus 123 would be encoded as either XQ or XV.

The reverse process would decode both XQ and XV to the Easting 123.

## ● Northings

In this case, the digit pairs range sequentially from 11 to 36. The encoding and decoding process works the same way as for Eastings. Again, there are multiple possible encodings for a Northing, but only a single decoding.

## ● Parameters

The only parameter for the ENC task is whether the first trial is encoding or decoding.

## ● Output Data

The Encoding task uses the standard data format for the record header. The extra data fields at the end of the record are:

Response\_Time:

Time between display of code and subject hitting enter (in milliseconds)

Encode/Decode:

"E" if subject is changing numbers to letters. "D" if subject is changing letters to numbers.

Prompt:

The text to encode or decode

Subject's\_Answer:

The subject's encoded (or decoded) answer

Expected\_Answer:

The encoded (or decoded) version of the prompt (Note: the second and fourth letters in an encoding are not unique. Thus the subject's answer may be different from the expected answer and still be correct.)

Eastings:

A permutation of the 26 capital letters the random code string presented for the East co-ordinate. Only available on the first record of a block.

Northings:

A permutation of the 26 capital letters the random code string presented for the North co-ordinate. Only available on the first record of a block.

# GKN -- General Knowledge Quiz

## ● Description

The subject is presented with a simple question and two possible answers. They then indicate how confident they are in their answer.

## ● The detailed task events are as follows

1. A question is randomly selected from the list without replacement.
2. The question text is displayed by itself for one second.
3. The first answer alternative is presented on below the question on the left side of the screen and the second is presented on the right side of the screen. Note that the question file author controls the assignment of the correct answer to the left or right side by specifying it as the first or second alternative respectively.
4. The subject specifies their choice by clicking on the answer text.
5. A confidence scale is presented and the subject selects a confidence level.
6. The process is repeated after a one second delay.

The task exits when the last question has been dealt with or the task duration expires, which ever comes first.

## ● Parameters

The only parameters for the General Knowledge task is the name of the file containing the questions.

## ● Input Data

The question file is a plain ASCII file which can be created with any editor. Each entry takes four lines as follows:

1. Question Identifier This is transferred to the result file in the record that corresponds to the presentation of the question. It is solely to assist you in analysing the results. It can be any ASCII string. Avoiding blanks and especially tabs would probably simplify analysis.
2. Question Text The text of the question.
3. Left Answer Alternative The text of the answer that will appear on the left side of the display. Placing an "\*" in the first character position will designate this answer as correct.
4. Right Answer Alternative The text of the answer that will appear on the right side of the display. Placing an "\*" in the first character position will designate this answer as correct.

If both answers begin with "\*" the second will be considered correct.

Note that it is the author's responsibility to place the questions files in the proper directory. Locate this as follows: Open the SUSOPS group that you will be using and click on the Dispatcher icon

once. Go to the menu and select File Properties and look for the Working Directory field. Suppose that it was C:\SUSOPS. You should put your files in C:SUSOPS\QUESTION.

## ● Output Data

The General Knowledge task uses the standard data format for the record header. The extra data fields at the end of the record are:

Response\_Time:

Time between display of answers and subject making a choice (in milliseconds)

Confidence\_Time:

Time between display of question and subject clicking on a confidence button (in milliseconds)

Which\_Question:

Question identifier (line 1).

Subject's\_Answer:

Which answer the subject chose (0=left, 1=right)

Right\_Answer:

The correct answer (0=left, 1=right, -1=none specified)

Confidence:

50 to 100 % confidence level

# GK2 -- General Knowledge Task 2

## ● Description

The GK2 General Knowledge Task 2 presents subjects with a series of problems in which they have to select one of two alternatives based on a specified criterion. In each problem, the criterion is displayed as a prompt in the center of the screen. An example of a prompt is "Which is larger?" (in reference to population). Following a delay, two alternatives are displayed in the lower left and lower right portion of the display. The subject makes a selection by depressing the corresponding button (ie. left or right).

## ● The detailed task events are as follows

Step\_one:

Load problems from a question file according to the order of problem presentation.

Step\_two:

Select the first problem and display it on the screen.

Step\_three:

After a short delay, display two alternatives.

**Step\_four:**

The subject makes a selection by depressing the corresponding button. If the subject response too slow and the Maximum Response Time option is set, a "Too Slow" message appears on screen.

**Step\_five:**

If the Confident Level Frame option is not set to "NONE", a Confident Level Frame appears. The subject will indicate how confident s/he feels the answer is.

**Step\_six:**

After the Confident Level Frame disappeared, if the Feedback option is set, a feedback message appears and tells the subject that his/her answer is correct or not.

**Step\_seven:**

After the feedback message disappear, if any, repeats Step two.

## ● **Preparing General Knowledge Experiments**

First, the experimenter must create a file which contains all the problems for an experiment run. Each problem contains a difficulty level specifier, a prompt and two choices. The problems are grouped by difficulty level and they have to come in order. First Easy, then Medium and then Hard. This file is in ASCII format and must use .ASC as the file name extension.

Second, s/he converts the ASCII problem file into .TMP file by using the tool GKNCOMP. Notice that two files are generated after using GKNCOMP: a .BIN file and a .TMP file. Only the .TMP file is used in this version of GK2. Also, in some version of GKNCOMP, the .TMP file is renamed to .ASC file, please make the necessary changes. The GK2 task requires this file to run property.

Finally, in the third step, the Experimenter creates a parameter file (.PRM) which specifies the characteristics of the presentation. This file can be created using the parameter screen in Schedule Editor.

All .TMP and .PRM files should be located in the /Question directory. Problems can be added to or removed from the original ASCII problem file. The Experimenter must remember to regenerate a new .TMP file.

## ● **Parameters**

The parameters for the General Knowledge Task 2 are:

**Param.\_File:**

The name of the parameter file or a new name if creating a new parameter file.

**Quest.\_File:**

The name of the question file.

**Question\_Color:**

The color of the choices, the background display and the prompts.

**Question\_Display\_Time:**

Display the question for how long before the choices popup.

Max.\_Response\_Time:

If the subject uses more than this time to response, a "Too Slow" message appears. This parameter is valid only if the Maximum Response Time checkbox is set.

Display:

Duration of the "Too Slow" message. This parameter is valid only if the Maximum Response Time checkbox is set.

Feedback\_Time:

Duration of the feedback message. This parameter is valid only if the Feedback checkbox is set.

Randomize\_Type:

Order of problem presentation. There are three types of orders: All Randomize - problems are selected at random from the entire file, Party Randomize - problems are randomized within groups and the groups are presented in the order Easy, Medium and Hard. None - problems are read and presented sequentially from the file.

Confid.\_Frame\_Type:

Type of Confident Level Frame. There are three types of frames: Full - Normal Confident Level Frame. Dummy - Same as the Full Confident Level Frame, but the subject's choice is not recorded. None - No Confident Level Frame.

Confident\_Level:

Minimum and maximum confident level, and the different between each generated confident levels.

## ● Output Data

The General Knowledge task 2 uses the standard data format for the record header. The extra data fields at the end of the record are:

Response\_Time:

This is the subject's response time in milliseconds.

Question\_ID:

The ID for the question that was asked.

Difficulty\_level\_with\_the\_optional\_Tag:

The difficulty level for the problem as coded in the general knowledge problem file with the optional tag specified by the experimenter.

Subject's\_Response:

This is coded as 0 if the subject selected the left choice and 1 if the subject selected the right choice.

Score:

This is coded as 1 if the subject responded correctly and 0 if incorrectly.

Confidence\_Level:

The confidence level as coded on the confidence button selected by the subject. A value of 0 is stored if the subject selected the "Error" button. A value of -1 is stored if the "Dummy" screen was shown, and a value of -2 is stored if the blank (None) screen was shown.

Confidence\_Response\_Time:

The time it took the subject to select a confidence level in milliseconds.

# GTP -- General Text Presentation

## • Description

The General Text Presentation Task (GTP) displays a series of prompts on the screen and records the subject's response to each. Prompts are formatted strings read from a file specified on the command line and can be one or more lines long. The subject can respond by mouse or keyboard as described below.

First\_example

A question is displayed on screen asking the subject "How do you feel?". A list of four choices numbered one to four follows. The subject then answers with his choice from the list.

Second\_example

The subject is asked to close his/her eyes. After a few seconds, a bell rings signalling the subject to open his/her eyes. A question is displayed on screen asking how much time has passed. The subject's response will be a number typed on the keyboard. Another group of text appears on screen. Previous text can either be erased or left on screen, depending on user's choice.

## • The detailed task events are as follows

1. Read a line from the source file (see Parameters) and separate it to up to five tokens -- Format code, Modifier code, Text1, Special code and Text2. More information on these five tokens follows.
2. Process Format code and create corresponding effect.
3. Process Modifier code and create corresponding effect.
4. Display Text1 on screen.
5. Process Special code and create corresponding effect.
6. Repeat steps one to five, until you obtain a Special code that halts the task and causes it to wait for the subject's response.
7. Repeat steps one to six after subject responds until you get a Format code that terminates the task or until all of the file has been processed.

## ● Parameters

The parameters for the General Text Presentation Task are:

Question\_File's\_Name

Name of a plain ASCII file with extension ".gtp" which contains prompts.

Tag\_ID

An identifies that is copied to the result file to aid in analysis.

## ● Input Data

The question file is a plain ASCII file which can be created with any editor. Each line in this ASCII file contains up to five tokens. Example:

```
0<CEN><BEL>How are you?/ANSWER
```

- Format Code is the first character of every line. In the example above, "0" is a Format code. If it is not recognized, it is treated as a SPACE. The following is a list of Format codes.
  - "\*" Clear screen first and start line at the top left corner.
  - " " Print Carriage Return, Line Feed(CR/LF) after line.
  - "1" Cursor remains at the end of text.
  - "2" Print Carriage Return after line.
  - "0" Clear line after handling special code(see below).
  - "+" Clear text1 and print CR/LF after handling special code(see below).
  - "-" Clear text2 and print CR/LF after handling special code(see below).
  - "X" Terminate this task.
- An optional list of Modifier codes can be placed after Format code. Each Modifier code is surrounded by "<" and ">". In our example, <CEN> and <BEL> are both Modifier codes. Here is a list of Modifier codes:
  - <BUT> Memorize the first non-blank character in Text1. When the task is waiting for the subject to respond, this non-blank character is one of the choices offered to the subject. It is displayed as a button on the screen.
  - <BEL> Signals subject by a beep sound.
  - <CEN> Displays Text1 at the center of current line.
  - <CLA> Clears line above and leaves cursor at the beginning of line above.
  - <CBL> Clears storage of non-blank characters memorized using <BUT>.
- Text1 is any text string located after Modifier code (see below). In the example, "How are you?" is considered as Text1.
- Special code is an optional code follows Text1. In our example, "/" is a Special code. Here is a list of Special codes:

- "/" Wait for up to N characters to be typed, where N is the length of text2(see below). These characters will be echoed.
- "%" Wait for a single character response. This character must be one of the characters in text2(see below). If text2 is blank, any character will be accepted.
- "^" Same as "%" except response must be one of the characters memorized by using <BUT> (see modifier code).
- "&" Wait for the time specified by text2. Text2 must be a standard VMS delta time format. If an inaccurate time format is used there will be no time delay.

5. Text2 is an optional text string that follows special code (see below). In the example, "ANSWER" is considered as Text2.

Therefore, our example will mean the following: Display "How are you?" (Text1) at the center of current line (Modifier code <CEN>). Signal the subject with a beep sound (Modifier code <BEL>). Task then wait for the subject's response, which is an answer of six characters or less (Special code and Text2). After the subject responds, erase the whole line. (Format code)

Note that it is the author's responsibility to place the questions files in the proper directory. Locate this as follows: Open the SUSOPS group that you will be using and click on the dispatcher icon once. Go to the menu and select File Properties and look for the Working Directory field. Suppose that it was C:\SUSOPS. You should put your files in C:\SUSOPS\QUESTION. Furthermore, your files must have ".gtp" as extension.

## ● Output Data

The General Text Presentation Task uses the standard data format for the record header. The extra data fields at the end of the records are:

Elapsed\_time

Subject response time in milliseconds.

Tag\_ID

The same Tag ID from parameter which indicates what type of question file is running.

Expected\_Answer

Text2.

Subject\_Answer

Response that subject provides.

# INN -- Instruction

## ● Description

The instruction task displays some text to the subject for a given amount of time. This text is usually instructions on how to do the task which follows. The text comes from a file.

If the instruction task is not given a filename it displays the word "BREAK".

## ● Parameters

The only parameters for the Instruction task is the name of the file containing the text.

## ● Input Data

The instruction file is a plain ASCII file which can be created with any editor.

## ● Output Data

The Instruction task produces no output at all.

# LRT -- Logical Reasoning

## ● Description

The Logical reasoning task presents a series of problems concerning the relationship between two entities A and B. A proposition is displayed of the form entity relationship other-entity. Where relationship can be one of the following:

1. precedes (coded as 0)
2. is preceded by (coded as 1)
3. does not precede (coded as 2)
4. is not preceded by (coded as 3)
5. follows (coded as 4)
6. is followed by (coded as 5)
7. does not follow (coded as 6)
8. is not followed by (coded as 7)

Examples would be A precedes B, and B is not followed by A.

The proposition is followed by a statement of the form "AB" or "BA" and the subject responds by clicking on either the TRUE or FALSE button.

Propositions and statements are generated randomly with replacement and are displayed concurrently. The task continues until the specified duration elapses.

## ● Parameters

The Logical Reasoning task has no parameters.

## ● Output Data

The Logical Reasoning task uses the standard data format for the record header. The extra data fields at the end of the record are:

Response\_Time:

Time between display of the proposition and the subject making a choice. (in milliseconds)

First\_Statement\_Character:

"A" or "B" from the statement AB or BA respectively.

Operation:

Codes for various relations such as "does not precede".

Second\_Statement\_Character:

"B" or "A" from the statement AB or BA.

First\_Proposition\_Character:

Eg., "A" from "A follows B".

Second\_Proposition\_Character:

Eg., "B" from "B follows A".

Expected\_Answer:

Is proposition True (T) or False (F)

Subject's\_Answer:

Did subject click on True (T) or False (F)

Note that the above ordering is counterintuitive. Sorry, but it is historical and we've got a lot of data.

## QST -- Questionnaire

### ● Description

The Questionnaire Presenter is a straightforward program that presents a questionnaire and records the answers. It does not attempt to record when, or in what order questions were answered.

Like all PenSusops tasks it will time out after a period specified in the schedule. Typically you will want to let the subject finish the questionnaire, rather than time out. Thus you should leave a generous amount of time in the schedule. The task operates as follows:

The questionnaire will be presented one page at a time. The subject can answer questions by using keyboard or by using mouse. Keyboard user may move to a box or a slider with Tab key or arrow keys, then check or uncheck the box with space bar or change the slider value with arrow keys. Word answers is typed into edit boxes. Mouse user may click on a box or a slider.

If there is a Next Page button then the subject can flip to the next page by pressing the button. Otherwise the questionnaire will automatically flip to the next page after all questions on the

current page have been answered.

If there is a Prev Page button then the subject can flip back a page, and review answers to previous questions (possibly changing them).

When the subject completes the last page the questionnaire is finished. The Next Page button (if present) will read Done on the last page to signify this. The subject can also terminate the questionnaire at anytime by pressing a "secret" task termination hot key. This hot key is ALT-SHIFT-F9.

## ● Output

When a questionnaire is presented to a subject the subject's answers are stored in a results file. This has much the same format as that of other PenSusops tasks. The major difference is that the header data is not repeated on each line.

The header line consists of the standard header data, followed by the filename of the questionnaire. The elapsed time on this record is the total time taken by the subject to complete the questionnaire. In addition, the last column of this header line is a word that indicates the questionnaire is terminated normally or is terminated by pressing the termination hot key. "SUCCESS" indicates a normal termination, while "ABORT" indicates a termination by the hot key.

Each subsequent line contains the answer to one question. The first two fields identify the question by page number, and question number within the page.

The last column is the subject's answer. If the subject did not answer the question then the value is "." (a single dot). If the subject chose "Not Applicable" then the answer is -9999. Otherwise the answer depends on the value of Leftmost Choice. By default this is 1, so the answers are on a scale from 1 to the Number of Choices. It can however be any value from -9998 to 30000.

# SRT -- Serial Reaction Time

## ● Description

The Serial Reaction Time task displays a four key keypad with a different character or symbol on each key. One of the four graphics will be shown in a display area and the subject must click on the corresponding key as quickly as possible. Display characters are selected from the four keypad characters randomly with replacement. Since this allows for sequential repetition of a character, the display colours are reversed from presentation to presentation so that the subject can detect the onset of a new stimulus.

## ● Parameters

The task accepts four parameters, a font name (which may include blanks), and the four keypad characters. The Schedule Editor provides a list of font names and displays the characters that you select (from your ASCII keyboard) in the selected font.

You should try to stick to the standard set of True Type fonts that are distributed with Windows. Fonts are not included with PenSUSOPS distribution disks for copyright reasons. Consequently, the task may produce an unexpected display if the machine that you run the schedule on does not have the specified font.

## ● Output Data

The Serial Reaction Time task uses the standard data format for the record header. The extra data fields at the end of the record are:

Response\_Time:

Time between display of the symbol and the subject hitting a button (in milliseconds)

Expected\_Answer:

the number of the correct button

Subject's\_Answer:

the number of the button hit by the subject

Font:

System font in which symbol is displayed

Symbols:

the symbols displayed on the four buttons

# STM -- Short Term Memory

## ● Description

A series of digit strings are presented to the subject along with a direction indicator. If the indicator is "FORWARD" the subject must enter the digits in the order they were presented. If it is "REVERSE", they must enter the digits in the reverse order. The digit string length and direction of recall vary according to a set of rules. The initial recall direction is FORWARD and the initial string length is four digits.

The detailed task events are as follows:

1. A digit string of the required length is randomly generated from the set 1..9 with replacement.
2. The recall direction, "FORWARD" or "REVERSE" is displayed throughout the presentation.
3. The digits are displayed one at a time for one second each with a half second delay between digits.
4. The keypad is enabled. The subject enters the digits by clicking on the digit buttons or by pressing corresponding keys on keyboard. When completed, click on the "ENTER" button or press the enter key. There are "CLEAR" and "BACKSPACE" editing keys available.

5. Subject input is aborted if they take too long. The allowed duration is Minimum Timeout + (string length - 4) \* Timeout Increment or Maximum Timeout whichever is less.
6. If the subject answered correctly, the length of the digit string is increased by one up to 30. If it reaches thirty, the recall direction is reversed.
7. If the subject answered incorrectly, the number of errors for the current direction is incremented. If it reaches Error Count, the recall direction is reversed, otherwise the length of the digit string is decremented to a minimum of four and the recall direction is unchanged.
8. Each time the recall direction is reversed the string length is set back to four and the current number of errors is set to zero.
9. The task terminated when the specified duration has elapsed.

## ● Parameters

The Short Term Memory Task uses four parameters:

Minimum\_Timeout:

Time allowed to enter the shortest length string (seconds).

Maximum\_Timeout:

Maximum time allowed regardless of allowance for string length (seconds).

Timeout\_Increment:

Additional time allowed for each digit in excess of four digits (seconds).

Error\_Count:

The number of errors tolerated before the recall order is reversed.

## ● Output Data

The Short Term Memory task uses the standard data format for the record header. The extra data fields at the end of the record are:

First\_Digit\_Time:

Time between unfreezing of keyboard and first digit typed (in milliseconds)

Last\_Digit\_Time:

Time between unfreezing of keyboard and last digit typed (in milliseconds)

Enter\_Time:

Time between unfreezing of keyboard and subject typing Enter (in milliseconds)

Expected\_Answer:

The digits that the subject must recall

Answer\_OK:

True (T) if subject correctly recalled the digits

Forward/Backward:

F if the digits were displayed in forwards order, B if the digits were displayed in reverse order

Subject's\_Answer:

The digits that the subject typed

Min\_Time:

Time limit for answer to shortest digit string (in seconds)

Max\_Time:

Time limit for answer to longest digit string (in seconds)

Time\_Increment:

Time limit increment for each digit greater than shortest string (in seconds)

Error:

Number of subject errors before program switches to forwards/backwards display of digits

## SUB -- Subtraction

### ● Description

The subject is presented with an initial subtraction problem. If Toughness is "EASY", the subtrahend is in the range 500...999 and the minuend is in the range 5...9. If it is "HARD", the subtrahend is in the range 500...9,999 and the minuend is in the range 12...36, with 22, 30 and 33 being excluded. The subject enters their answer by clicking on a numeric keypad or pressing the corresponding keys on the keyboard. Each time the required number of digits is entered the display is cleared (thus the subject can not edit the last digit). The subject continues by repeatedly subtracting the fixed minuend from their most recent result. No further input is accepted after the specified duration has elapsed. The task terminates after the specified rest period expires. The rest period allows several SUB tasks to be run with short durations sequentially in a schedule.

### ● Parameters

The Subtraction Task uses two parameters:

Toughness:

Whether the subtractions are easy ("E") or hard ("H").

Rest\_Period:

The number of seconds in the rest period at the end of the task (seconds (MAX 65)).

### ● Output Data

The Subtraction task uses the standard data format for the record header. The extra data fields at the end of the record are:

Response\_Time:

Time between display of the prompt and the subject entering the final digit (in

milliseconds).

Expected\_Answer:

Running total of subtractions assuming perfect performance.

Minuend:

The minuend from the initial problem.

Subject's\_Answer:

Subject's answer.

Difficulty:

E if subtraction problem is easy, H if hard.

## SYN -- Synwork Launcher

### ● Description

The Synwork Launcher will run the stand alone version of SYNWORK. The experimenter is responsible for specifying a command line, along with arguments required, to run SYNWORK. The SUSOPS duration control will override the duration argument of SYNWORK.

### ● The detailed task events are as follows

Starts SYNWORK from SUSOPS.

The subject will complete the task.

Go back to SUSOPS when the subject completed the task or the time is up.

### ● Parameters

The parameter of the Synwork Launcher task is a command line to run SYNWORK, along with correct arguments. Full path of the SYNWORK executable file is required. For example:

```
instead of using  
synwork1 /D10
```

```
the experimenter should use, for example,  
c:\synwork\synwork1 /D10
```

The duration parameter(/Dn) will be overridden by the SUSOPS duration control. As a result, even the experimenter has specified /D10 (10 minutes duration) as an argument, it will not affect the task's duration unless the experimenter sets the SUSOPS duration control to 10 minutes as well. On the contrary, if the experimenter sets the SUSOPS duration control to 10 minutes, while without specifying a duration argument in the command line, the task will still run properly. Also notice that this task's duration is minute-based. All duration setting will be rounded to the closest minute.

## ● **Output Data**

The Synwork Launcher task runs SYNWORK only, and therefore, it will not generate any result by itself. All the experiment data are recorded by SYNWORK and are stored in their usual formats.

# Questionnaire Editor

## Description

The Questionnaire Editor allows the experimenter to create and edit multiple-choice and short answer questionnaires. This is done with a visual approach, so the editing session looks much like the final product.

## Topics

- [Basic Concepts](#)

A questionnaire consists of questions, text and images. Text items are purely decorative, and are used for titles, and instructions. Image items are similar to text items, except they display images rather than text. Questions have text too, but they also have a chooser which is used to answer the question. The chooser may be a multiple-choice list, a scroll bar or a short answer. The question may also have labels which are used to identify values of the chooser. Typical labels might be "Agree", "Don't Care", and "Disagree".

- [Quick Tutorial](#)

To get a quick idea of the editor you should try it. Start it by clicking on the Questionnaire Editor icon in Windows.

- [Menu Commands](#)

There are three menus in the questionnaire editor. The menu items are listed below.

- [Main Edit Window](#)

Most of the editing is done on the main window. This window shows a page full of questions in the same format that the subject will eventually see them in. There are a few differences:

- [Text Edit Window](#)

You arrive at this window by Double-Clicking on a text box in the main window. It lets you change the attributes of that text box. If the text box is part of a question then you can also change some attributes of the chooser as well.

- [Image Selection Window](#)

To display the image selection window, Double-Click an image item in main window. This new window allows image selection in the image item double-clicked. Image file must be in BMP format.

- [Choose Edit Window](#)

If you Double-Click on a chooser in the main window then you will get this window. It lets you change the attributes of a single chooser. There is no Chooser Edit Window available for edit box.

- [Questionnaire File Format](#)

In order to make the loading time fast the questionnaire file has been written in a machine readable format. This means that it is not convenient to edit it with any tool other than the Questionnaire Editor.

## Basic Concepts

### Description

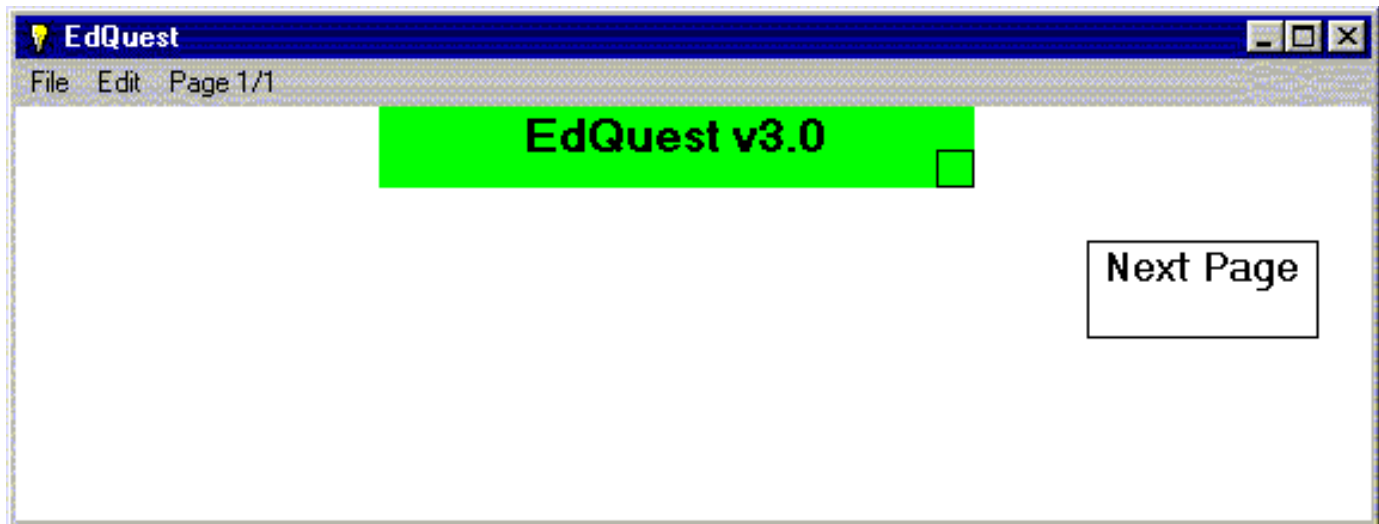
A questionnaire consists of questions, text and images. Text items are purely decorative, and are used for titles, and instructions. Image items are similar to text items, except they display images rather than text. Questions have text too, but they also have a chooser which is used to answer the question. The chooser may be a multiple-choice list, a scroll bar or a short answer. The question may also have labels which are used to identify values of the chooser. Typical labels might be "Agree", "Don't Care", and "Disagree".

A questionnaire can have multiple pages. It will be presented to the subject a page at a time. Each page may have a Next Page and a Prev Page button. These let the subject move between pages. If there is no Next Page button then the subject is moved to the next page upon answering all the questions on the current page.

# Quick Tutorial

## Description

To get a quick idea of the editor you should try it. Start it by clicking on the Questionnaire Editor icon in Windows.



When the Editor opens you will see a single text item that says something like "EdQuest v3.0".

In the lower right corner of the text box you will see a small square that can be used to resize the box. Put the pen down in that square and Drag it around the screen.

The box can be moved around the screen by placing the pen down somewhere inside the text box and dragging.

Double-Click in the text box to get a new window. Clicking on the text in the new window will allow you to type new text (use the delete key to erase old text). When you are finished Click the OK button on the right side of the new window. You will see the text appear as it will look on the questionnaire. (If your new question is too long you may have to resize the box so that you can see it all.)

To add a new image, go to the Edit menu and choose Add Image. A new image box appears. At this point, there is no image in the image box. Double-Click in the image box and you will see a new window appear on screen. You may choose image file from this window. When you are finished, Click the OK button on the right side of the new window. Resize and move the image box the same way as a text box if necessary. There is no resizing box in the image box. When the cursor is moved to the lower right corner, it will change into a resizing cursor.

If you go to the Edit menu and chose Add Question you will see a new question appear below the first one. You can do this several times to make multiple questions, and the text of each question can be changed at any time.

When you are happy with this page you can go to the Page menu and pick New Page. You will see an empty page. You will also notice that the Page menu now says "2/2" rather than "1/1". The second number is the number of pages that exist, and the first number is the page that you are currently looking at.

To copy a question to a new page pick Prev Page from the page menu to go back to the first page. (The Page

menu will now say "1/2".)

Select the question by Clicking on its text (the question will darken. If you have a colour screen you will see it change from green to blue). Now choose Copy from the Edit menu.

Choose Next Page from the Page menu to go back to your new page. Now click on Paste from the Edit menu to copy the question to the new page.

Play around some more if you like, or read the detailed description of the available features in the sections below.

# Menu Commands

## Description

- There are three menus in the questionnaire editor. The menu items are listed below.

## File Menu

The file menu is used to save and restore questionnaires. The commands contained in it are listed here.

### New File

deletes all existing questions, and gives you a new questionnaire.

### Open

asks for the name of a questionnaire file and reads it in. (This replaces the current questionnaire.) It will open to the first page.

### Merge In

also reads in a file, but it is appended to the end of the current questionnaire. The cursor will be on the first page of the new material that you have read in.

### Save

saves the questionnaire under the current filename (i.e. the last one you read from or saved to). If the editor does not have a current filename it will ask for one. (You can see the current filename in the title bar of the editor.)

### Save As

always asks for a filename. It then saves the questionnaire in the same manner as Save.

### Print Design

prints design layout of the questionnaire on the default printer. Every question in design layout contains a "Tag" ID. This ID is corresponding to the ID on each question's result.

### Print Presentation

prints presentation layout of the questionnaire on the default printer properties. Questionnaire can be presented to the subject by using a copy of presentation layout instead of using computer.

### Printer Setup

allows default printer selection and alteration of printer.

### Quit

lets you exit the program. If you have unsaved changes to your questionnaire you will be asked if you want to save them.

## Edit Menu

The edit menu lets you add and modify questions, text and images.

### Add Question

will create a new question. It will be a copy of the last question on the page.

### Add Text

will create a new text item. It will be a copy of the last text item on the page.

### Add Image

will create a new image item. A new image item will not contain any image initially.

### Cut

will delete all the selected questions, text items and image items. (You select a question, text item or image item by clicking on it.) The deleted questions, text items and image items are saved in the cut-and-paste buffer.

### Copy

will copy the selected questions, text items and image item to the cut-and-paste buffer.

### Paste

will put questions, text and image from the cut-and-paste buffer into the current page. The cut-and-paste buffer remembers where the question used to be, and will try to paste it back to the same location on the page. If that place is occupied then the new question will be placed lower on the page.

### Undo

will undo the last change.

### Change Order

allows you to shuffle the order of questions on the page. If you look at the left side of the chooser for a question you will see a number. The first question is numbered "1" and so on. This number has no effect on the presentation of the question to the subject. It is only used in the output file, where it identifies which question has been answered.

When you choose Change Order you will be asked for a new number for the question. Other questions will be shuffled out of the way to make room for it. You may have to use this command a few times to get all the questions in the correct order.

## Page Menu

The page menu lets you manipulate pages and buttons. The title of the page menu shows the numbers of the current and maximum page (i.e., 2/5 means you are on page 2 of a 5 page questionnaire).

### New Page

creates a new (empty) page, and moves you to it. Typically you will want to copy some items from the previous page, and paste them on the new page.

### Prev Page

moves you to the previous page. If you are already at the first page it does nothing.

### Next Page

moves you to the next page. If you are at the last page it creates a new one (just like New Page). If you don't want this page you can delete it.

### Goto Page

asks which page you wish to go to. It will then jump to that page.

### Delete Page

deletes all the questions on the current page and shuffles any following pages up to fill the gap. It warns you first if there is anything on the current page.

There is no corresponding "Insert Page" command.

#### Allow Loitering

makes the "Next Page" button visible. When the questionnaire is presented to the subject a page without a "Next Page" button will be finished as soon as the subject answers all the questions on the page. If there is a "Next Page" button then the subject must click on it to move to the next page.

The menu item will have a checkmark beside it if the "Next Page" button is visible.

#### Allow BackTrack

makes the "Prev Page" button visible. If it is there then the subject can back up and reconsider the answers to questions on the previous page of the questionnaire.

It is impossible to put a "Prev Page" button on the first page, or on a page which does not have a "Next Page" button.

## Main Edit Window

### Description

Most of the editing is done on the main window. This window shows a page full of questions in the same format that the subject will eventually see them in. There are a few differences:

- \* The text, choosers, and labels will have a green background so that you can see the areas that their boxes take up. Selected text has a blue background.
- \* The text also has a small square in the lower right corner. This is the resize area (described below).
- \* Choosers have a number in the top-left corner. This is the question number. (See [Change Order](#) above.)

### Question Text and Text Items

The text of a question and the text of a text item look alike and act in the same way. They will be called text boxes in the following description.

If you Click on a text box it will be selected. This lets you Cut or Copy it (see above). A selected question changes colour to blue rather than the normal green.

If you hold down the Shift key while clicking on a text box then it will be selected, but other questions will not be de-selected. This lets you select multiple questions. Doing another shift-click on the question will de-select it again. Multiple selections are only useful for cut and copy commands. You can't drag or resize multiple questions at once.

If you hold down the pen and move it you will Drag the text box. If it is a question then the chooser and labels will disappear, and will appear in their new location when you finish moving.

You will note that there is a small square in the lower-right corner of the text box. If you hold down the pen and drag here you will Resize the text box.

If you Double-Click in the text-box (i.e. click twice rapidly) you will open the Text Edit window (described below). This lets you change the contents and attributes of the text.

## Image items

Image items have the same "Select", "Move" and "Resize" features as text items. There is no resizing box in the image box. When the cursor is moved to the lower right corner, it will change into a resizing cursor.

## Choosers

Every question will have a response object. This is known as the chooser. There are currently three types of chooser: multiple choice, slider and edit box. You can change the chooser type in the text Edit window (see below).

Choosers are "stuck" to questions. By default they are stuck to the top-left corner. When a question is moved or resized the chooser is relocated to be in the proper position relative to the text. You can change where the chooser is stuck in the text Edit window (see below).

If you Drag a chooser you can move it. Generally you will want to drag it right or left if the chooser is stuck to the side of the text box, and up or down if it is stuck to the top or bottom.

If you Double-Click on a chooser you will go to the Chooser Edit window (see below) where you can change some of the options.

Choosers may have labels for their choices. These will follow the chooser if it moves. They can only be edited in the Chooser Edit window (see below).

## Prev Page and Next Page Buttons

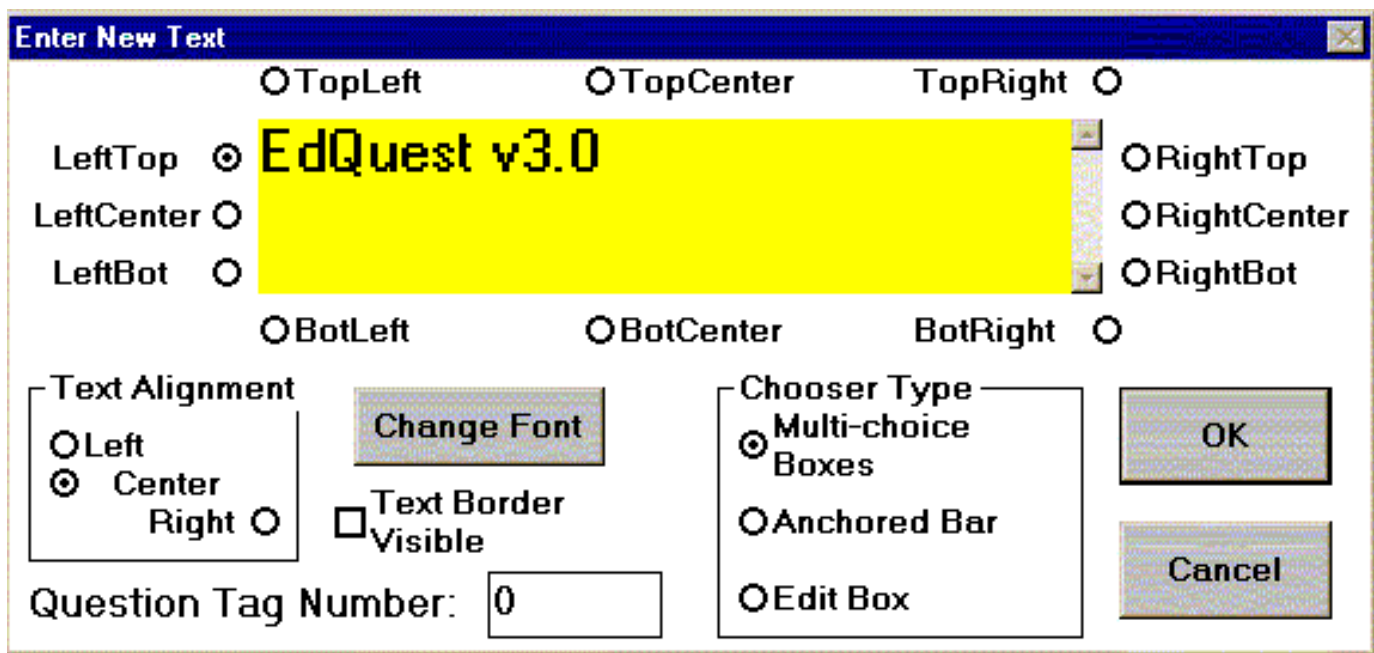
The Prev Page and Next Page buttons appear on the window if they are chosen from the Page menu (see above).

If you Drag the buttons you can move them around the screen. They can be in a different place on the screen for each page, but doing this may confuse the subject.

# Text Edit Window

## Description

You arrive at this window by Double-Clicking on a text box in the main window. It lets you change the attributes of that text box. If the text box is part of a question then you can also change some attributes of the chooser as well.



## Text

In the center of the window is the text. You can click on it and type in more text. The delete and backspace keys let you remove old text.

You can put a carriage-return in the text by holding down the control key while pressing the enter key. If you enter a lot of text then some of it may scroll off the top of the box. The up-down and right-left arrow keys let you move around in the text.

## Text Alignment

In the lower-left corner are a set of buttons to let you choose whether the text is centered, or left or right aligned. Unfortunately the change will not show up on this window, but you can see it when you go back to the main window.

## Change Font

The Change Font button allows you to pick a new font (like Helvetica) or to change the size or style (bold, italic underline and strikeout). These changes will show up on this window.

## Text Border Visible

This button allows you to outline the text box. This can make a nice visual effect.

## OK and Cancel Buttons

The OK button saves all your changes and returns to the main edit window. The Cancel button tosses away all your changes and returns to the main edit window.

## Question Tag

The Tag value is written out to the results file along with the answer to the question. You can use this to uniquely identify each answer.

## Stick to

If the text box is part of a question then the Stick to buttons appear. They let you choose which part of the text box the chooser is stuck to. For example if you pick BotCenter then the chooser will be placed below the center of the text box. If you resize the text box then the chooser will automatically move to the new center.

You can also drag the chooser relative to this point (see Chooser Drag above). It is quite possible that you could drag the chooser far to the left of the text box, and then drag the text box to the left of the screen. If this happens you will not be able to see the chooser any more, and thus you can't drag it back. To fix this problem just choose another of the Stick to buttons. This will place the chooser back beside the text box.

## Chooser Type

If the text box is part of a question then the Chooser Type buttons appear. They let you change the type of chooser. The three current types are multiple-choice, scroll bar and edit box.

# Image Selection Window

## Description

To display the image selection window, Double-Click an image item in main window. This new window allows image selection in the image item double-clicked. Image file must be in BMP format.

## Drive, directory and file list

Locate image files with the drive list, the directory list and the file list. You may select an image file by single click the file name, followed by a click on the OK button; or double-click the file name.

## OK and Cancel buttons

If you have selected an image file, click the OK button. To cancel, click the Cancel button.

# Choose Edit Window

## Description

If you Double-Click on a chooser in the main window then you will get this window. It lets you change the attributes of a single chooser. There is no Chooser Edit Window available for edit box.

## Current Label

The labels appear at the top of the window. Most have a green background. The current label has a blue background. Click on any label to make it the current one. Its text will appear in the text box below.

## Edit Label

Click in the text box to edit the current label. The label will update each time you type a key. Hit the enter key to make a multi-line label.

## Clear Labels

In the bottom center of the window is the Clear Labels button. If you Click it all the labels will be erased.

## Reset Labels

Below this is the Reset Labels button. If you Click it the labels will be changed to 1, 2, 3, ... N/A. These are the default values of the labels.

## Leftmost and RightMost Choice

On the lower left is the Leftmost Choice box. To change it Click on the box and type in a new value.

Most multiple choice answers start at 1 (as in "rate the difficulty on a scale from 1 to 10"). The Leftmost Choice lets the scale start at zero (or at any other number from -9998 to 30000).

Since the increments on sliders are not restricted to integers both the RightMost and LeftMost values are used to set the range of the slider.

This will affect the output from the questionnaire. It will also change the default labels produced by the reset

- labels button (see above).

## Number of Choices

Above the Leftmost Choice box is the Number of Choices box. To change it Click on the box and type in a new value.

- This affects the number of choice boxes that appear. The default is 5 and the maximum is 11.

## N/A Box

To the lower right is the N/A Box. To change it click on the box. This adds a "Not Applicable" choice to the question. It is slightly separated from the other choices to distinguish it.

- 

## No Feedback

- Available only for scroll bar. It provides option for not showing scroll bar's value.

## Scale Feedback

Available only for scroll bar. If selected, the precision of the displayed feedback value is limited to the number of decimal places specified in "Decimal Places" (See below).

- 

## Decimal Places

Available only when Scale Feedback box is selected. It specifies the number of decimal places in scroll bar's value.

- 

## Outline Labels

At the bottom right is the Outline Labels Box. To change it click on the box.

- This outlines each of the labels in a small box.

## Resizing Boxes

In the center of the window is a drawing of the multiple-choice boxes. Dragging them will affect the size and spacing of the boxes. Dragging right will space the boxes out wider. Dragging left will scrunch them closer together. Dragging up will make the boxes smaller. Dragging down will make them larger.

The program keeps you from making the boxes overlap. If you push them too close it will choose a smaller box size.

- 

## OK and Cancel Buttons

On the right side of the window are the OK and Cancel buttons.

The OK button saves all your changes and returns to the main edit window. The Cancel button tosses away all your changes and returns to the main edit window.

# Questionnaire File Format

## Description

In order to make the loading time fast the questionnaire file has been written in a machine readable format. This means that it is not convenient to edit it with any tool other than the Questionnaire Editor.

# Distributing PenSUSOPS System

## Description

Typically, a Scientist creates and tests a schedule on their own computer but runs it on one or more others that will be used by subjects. Further, a Scientist might wish to send an experiment to a colleague in another laboratory.

These situations present a software configuration problem since a large number of programs and other files are needed in order to run a PenSUSOPS experiment. These include the tasks themselves, Windows and VisualBasic components, experiment schedules, and all of the support files (instructions, questionnaires, etc.) that the schedules reference. If any of these are not transferred to the subject's computer, a schedule that ran perfectly for the Scientist might fail for the subject.

A utility program, the Packer, was created to handle this problem. The packer automatically finds all of the components needed to author and run new PenSUSOPS experiments. This facility can be used to transfer the system to other researchers. As an option, experimental schedules can also be included in the distribution. If this is the case, all files referenced by the schedules will also be included. The Packer allows the Scientist to include comments or notes with the package. All of the files are compressed and transferred to floppy disk.

The distribution disk(s) contain a standard Windows Setup program that can be run from the File Manager or Program Manager. This will transfer all of the Windows and VisualBasic components to the Windows directory and the PenSUSOPS tasks and files to a drive and directory of the user's choosing. The Setup program also creates a program group on the desktop and displays the originator's comments.

Full user documentation for the Packer is available as online help. This can be accessed directly from the desktop or used in a context sensitive manner when the Packer is running.

## Topics

- [What is Included in a Distribution](#)

As described above, all PenSUSOPS tasks and required Windows and VisualBasic components are always included. In addition, certain instruction files are always included. Specifically, for all tasks in the SUSOPS Programs directory (eg., ADD.EXE), the matching instruction file (eg., ADD.INN) in the Instruct directory will be packed if it exists. This ensures that a default set of instructions is available for all test tasks. There is no need to modify these since you can create and use your own instruction files (with different names) if you wish.

- [Version Control Issues](#)

You can reinstall your version of PenSUSOPS into the same directory that you used when you originally ran the Setup program. You might want to do this if some of the distribution files were accidentally deleted or overwritten. Setup will overwrite files which are not newer than the ones on your system, and will not modify or delete any files on your system which were not part of the distribution. This means that scripts, instruction and questionnaire files, etc., that you added or modified will not be affected in any way.

# What is Included in a Distribution

## Description

As described above, all PenSUSOPS tasks and required Windows and VisualBasic components are always included. In addition, certain instruction files are always included. Specifically, for all tasks in the SUSOPS Programs directory (eg., ADD.EXE), the matching instruction file (eg., ADD.INN) in the Instruct directory will be packed if it exists. This ensures that a default set of instructions is available for all test tasks. There is no need to modify these since you can create and use your own instruction files (with different names) if you wish.

No other files are included unless you elect to include one or more schedules as part of the distribution. Furthermore, only those files needed to run the selected scripts will be included. The implications of these points are as follows:

Suppose that you created a number of instruction files, questionnaires and General Knowledge files. These might be useful in a variety of experiments and in fact, you created them as general experiment resources. If you create a distribution with no scripts to send to a colleague, none of these files will be included.

Suppose that you had created a number of schedules, each of which referred to a distinct subset of questionnaires, and you chose to include some of these schedules in your distribution. Only those questionnaires referenced in the selected schedules will be included in your distribution.

You can force the Packer to include the support files that you want to share. Suppose that you created a set of French language instruction files and some questionnaires. Use the Schedule Editor to create a "dummy" schedule, that is, one that would never be used for an experiment. In this schedule, include a series of instruction tasks, each of which refers to one of your French language instruction files. Do the same with questionnaire tasks, making reference to your questionnaire files. When you create your distribution, make sure to include the "dummy" schedule. It will cause the Packer to include the files that you want. While this seems circuitous, the alternative of packing up everything would simply propagate outdated and unused files. In fact, you can clean up your PenSUSOPS directories by simply creating a distribution with all of the schedules. All of the support files that you are using will be included. You can then back up your system, delete the PenSUSOPS directory and reinstall the system using Setup.

Note: The Packer will not pack up experiment results files. You are responsible for maintaining these yourself.

# Version Control Issues

## Description

You can reinstall your version of PenSUSOPS into the same directory that you used when you originally ran the Setup program. You might want to do this if some of the distribution files were accidentally deleted or overwritten. Setup will overwrite files which are not newer than the ones on

your system, and will not modify or delete any files on your system which were not part of the distribution. This means that scripts, instruction and questionnaire files, etc., that you added or modified will not be affected in any way.

You should always install new distribution disks into new directories. New distributions may contain new versions of tasks and instructions which might not operate the same as previous versions. For example, suppose that one of the test tasks is changed. If you installed the new distribution disk into an existing PenSUSOPS directory, the original task would be overwritten. This means that if you ran one of your scripts that used the task, new subjects might be presented with different stimuli and responses than were presented to previous subjects. Currently, the incremental cost of installing a second version on PenSUSOPS is less than 400Kb of disk space.

Finally, we strongly recommend that you do not add files to the PenSUSOPS "PROGRAMS" directory. This will have unpredictable effects for the Packer and Setup programs.

# Maintenance and Support

## Description

PenSUSOPS software and documentation was developed by NTT SYSTEMS INC. under contract to the Canadian Department of National Defence. It is copyright Her Majesty the Queen in Right of Canada as represented by the Minister of National Defence.

The system is being maintained by NTT SYSTEMS INC. Questions and comments may be addressed to:

Marc Grushcow, [Marc.Grushcow@NTT.ca](mailto:Marc.Grushcow@NTT.ca)

# About

## Description

About PENSUSOPS

