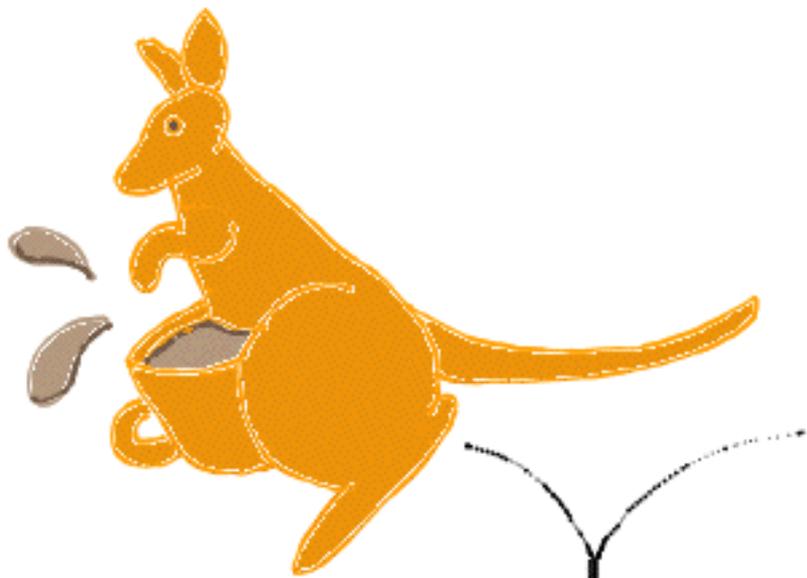


JWatcher™ 0.9
An Introductory User's Guide

**Daniel T. Blumstein, Christopher S. Evans
&
Janice C. Daniel**



JWatcher™

Animal Behaviour Laboratory
Macquarie University

<http://galliform.psy.mq.edu.au/jwatcher/>

Copyright © 2000–Daniel T. Blumstein, Christopher S. Evans & Janice C. Daniel.
All rights reserved.

Last modified: 14 November 2000

Table of Contents

1 A brief overview of JWatcher	3
1.1 Compatibility and Reliability	3
1.2 Installing JWatcher	4
1.3 Legal MumboJumbo	4
1.4 Acknowledgements	5
2 Using JWatcher	6
3 Global Definition	9
4 Focal Master	10
4.1 Test Details	10
4.2 Questions	10
4.3 Define Codes	10
5 Data Capture	11
5.1 Data File	11
5.2 Questions and Answers	11
5.3 Capture Data	11
5.4 Note on 'initialising a focal'	12
6 Edit Data	13
7 Focal Analysis Master	14
7.1 Modifiers	14
7.2 States	14
7.3 Exclusions	16
7.4 Time	16
7.5 State Analysis	18
7.6 Event Analysis	19
8 Combinations	21
9 Analysis	23
9.1 Analyse Data	23
9.2 Combine Key Codes	24
10 Strategies for using JWatcher	26
11 Warnings and errors	27
12 Frequently asked questions about JWatcher	30

1 A Brief Overview of JWatcher™

JWatcher is a powerful tool for the quantitative analysis of behaviour. It can be used to address any theoretical problem that requires a complex sequence of actions to be scored by a human observer. Thus, it can be used to study human or non-human behaviour. JWatcher is written in the Java™ computer language (<http://www.sun.com/java>) and is therefore not dependent on a particular operating system. It will run on virtually any microcomputer capable of running a Java Virtual Machine™, and it has been tested on both Windows™ (<http://www.microsoft.com>) and Macintosh™ (<http://www.apple.com>) operating systems.

The simplest use of JWatcher is as an event recorder that logs the time at which keys are pressed. It was designed primarily for focal sampling with continuous recording (i.e., a single subject is observed continuously for a predetermined time). The real power of JWatcher is, however, in its analysis routines. These calculate time budgets and statistics about the duration of behavioural states, and the intervals between them.

We developed JWatcher because existing event-recording programs typically force the user to make a restrictive series of decisions before beginning to observe behaviour. JWatcher imposes few such constraints. It is based on a 'score once, analyse many times' philosophy that we have adopted in our studies of animal behaviour. For instance, a single data file can be used to calculate time budgets over both short and long periods, or to track changes in different subsets of behaviours seen. Each behaviour can be analysed both as an instantaneous event (measuring frequency), and as a state (measuring total duration). The logical relationships between behaviours can be edited in post-processing (i.e., a given behavioural state can be redefined as independent or as mutually exclusive with others). The number of states that can be tracked independently is limited only by the number of alphanumeric keys available. Users may choose to exclude or ignore certain behaviours in analyses, and categories can be combined to generate behavioural codes that were not originally scored. Batch mode analyses allow efficient processing.

JWatcher was designed to be used in association with common word processing, spreadsheet, and graphics software. It produces simple output files that can readily be imported by such programs. We have concentrated on developing new scoring/analysis algorithms, rather than duplicating functions in existing software.

1.1 Compatibility & Reliability

JWatcher has been comprehensively tested but there are still some quirks in the way that it behaves on different computers and under different JVMs (i.e., Java operating systems). We have developed and tested it using the Mac OS Runtime For Java version 2.2.3, and with the Windows Java Runtime

Environment version 1.1.8. **You should only use these JVMs to run JWatcher.**

To our knowledge, the results are logically and numerically correct. If JWatcher behaves strangely, it is most likely a memory problem. JWatcher should be run on a machine with a minimum of 68mb free RAM. However, if you should experience memory problems, quitting and restarting the program (and quitting other open programs) should allow JWatcher to run more smoothly. JWatcher runs best on reasonably fast machines (i.e., >250 mHz).

We welcome bug reports and suggestions for new features via email to jwatcher@galliform.psy.mq.edu.au. We will attempt to fix bugs in future versions, but currently have no funding for future development.

Please note: While we have provided a FAQ that will be updated periodically, we are unable to personally respond to any queries about the routine use of JWatcher.

1.2 Installing JWatcher

You **must** have some version of Java installed on your computer in order to run the installation program. To reduce confusion later, we suggest first downloading the appropriate JRE/MRJ.

- For Windows systems, go directly to Sun Microsystems (<http://www.java.sun.com/Download4>) and download the JRE 1.1.8.
- For Macintosh systems, go directly to Apple Computer (<http://www.apple.com/java>) and download the MRJ 2.2.3.

Download the appropriate version of JWatcher from the JWatcher site (<http://galliform.psy.mq.edu.au/jwatcher/>) for your operating system. At this stage, we are providing a Windows and a Macintosh version. If you already have the Mac OS Runtime For Java version 2.2.3, or the Windows Java Runtime Environment version 1.1.8 you can download the versions without the JVM. Otherwise, you will have to download the versions that include the operating system specific JVMs.

Once downloaded, double-click on the installation icon. By default, JWatcher is installed into your Applications directory and will create an alias on your desktop. If you are running Windows, you will have to specify the JRE that JWatcher is to run with; be sure to select JRE 1.1.8.

1.3 Legal MumboJumbo

The program and manual are copyright by the authors, but JWatcher is 'freeware': this version is given away with no warranty expressed or implied.

Feel free to use it, teach with it, and tell others about it. We request that you tell others to download the program directly from the JWatcher web site (<http://galliform.psy.mq.edu.au/jwatcher>) to ensure they obtain the most recent version of the program and the manual. The authors and their respective institutions are not responsible for its use or misuse.

JWatcher is a registered trademark of the authors. Java, the Java Virtual Machine, and Sun, are registered trademarks of Sun Microsystems, Inc. in the United States and other countries. Windows and Internet Explorer are registered trademark of the Microsoft Corporation. Macintosh is a registered trademark of Apple Computer, Inc. Any other trademarks are the property of their respective owners.

1.4 Acknowledgements

For programming, we thank Fiona Walkerden, Xuhong Li and Derek Renouf of Adaptive Arts Pty Ltd (www.adaptive-arts.com), Linda Evans for creating the JWatcher icon, and members of the Animal Behaviour Lab for help with detailed testing. Initial JWatcher development was supported by a Macquarie University Research Infrastructure Block Grant. During initial development, Dan Blumstein was supported by an Australian Research Council Postdoctoral Fellowship, and Chris Evans by an Australian Research Council Large Grant.

2 Using JWatcher

To quantify behaviour you must create several types of files. The following description is an outline only to illustrate the overall structure and logical relationships (Figure 1). Detailed instructions are provided in the subsequent sections, and also in a series of on-line help pages.

Note: The on-line help requires your computer to have a web browser. If the help file does not load properly, try changing your default web browser to Microsoft Internet Explorer™.

First, you must create a Global Definition File (.gdf). Global definition files are a list of all the behaviours that you might possibly be interested in scoring. The .gdf file could be regarded as an ethogram—a catalogue of behaviour. Additionally, you can specify modifiers that may follow certain behaviours. Modifiers can be anything that has a logical relationship with a behaviour (e.g., the identity of the other individual in a dyadic social interaction, the location where a behaviour was performed, or a variable component of a display). The total number of behaviours and modifiers in a .gdf file can not exceed the number of alphanumeric characters on a keyboard.

When asking a focussed behavioural question, however, you are typically interested in a sub-set of behaviours and modifiers. You must specify the duration of a focal observation, and any question with which the user is to be prompted before beginning a focal observation (e.g., subject identity, date, time, etc.).

To do this, you will have to create a Focal Master File (.fmf). The focal master file first inherits a pre-existing global definition file (.gdf) to which you can add or subtract behaviours or modifiers. Additionally, the .fmf requires you to specify questions to be prompted with prior to scoring behaviour, to specify a focal duration, and to choose whether you want the event-recorder's clock to count up or count down when scoring behavior.

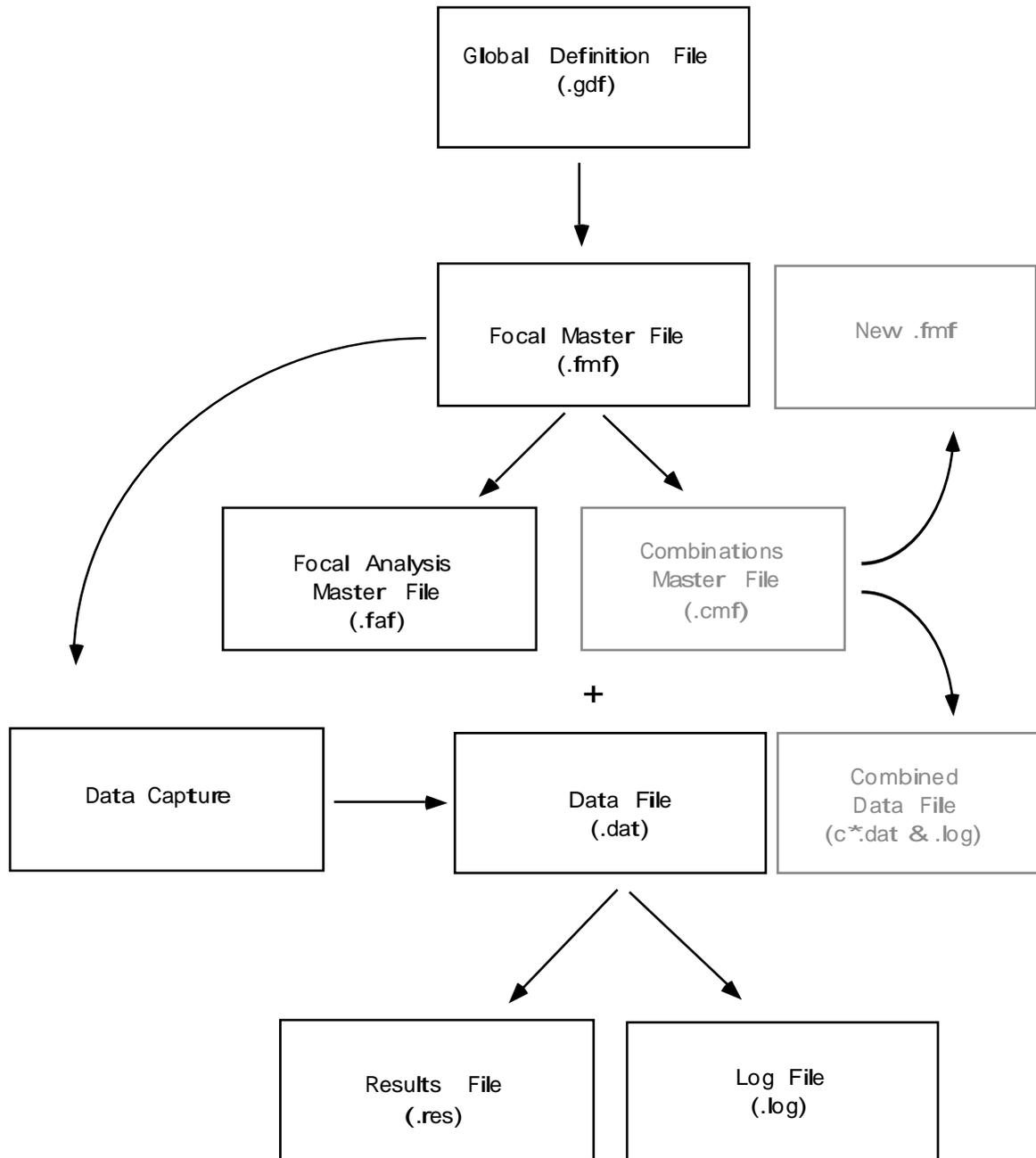
To score behaviour, the user must specify a focal master file (.fmf). The Data Capture routine allows you to score behaviour and then creates a data file (.dat) which contains a log of the key presses scored to the millisecond.

Before analysing a data file (a .dat file), the user must create a Focal Analysis Master File (a .faf file). The focal analysis master file is the 'guts' of JWatcher and specifies which behaviours are to be treated as states and which are to be treated as events. It also allows the user to 'time window' a single longer focal observation and summarise behaviour in each time window. It allows the user to exclude or modify certain behaviours and a variety of other specific functions. Once a .faf file is created, JWatcher can analyse a single data file or it can batch process all data files contained in a folder. JWatcher results files (.res) are comma delimited text files which are

opened easily in spreadsheet programs for subsequent post-processing and analysis.

Sometimes, users may wish to score behaviour in some detail and later combine detailed key codes into a single behaviour type. For instance, a subject may 'look' in a variety of postures—it can stand and look, crouch and look, or rear-up and look. JWatcher contains a routine to combine these different key codes into a single key code—look. To do so, a Combinations Master File (.cmf) must be created. Once a .cmf is created, JWatcher can combine the codes in either a single file, or it can batch process many files. The files resulting from combination (c.dat files) are analysed by a modified .fmf and a new .faf. Both of these files are based on the new key codes defined in the .cmf.

Figure 1. A taxonomy of JWatcher files and their relationships.



3 Global Definition

Global Definition is a comprehensive ethogram you will create for a given species or project. An ethogram is a catalogue of behaviours and modifiers specified, typically, at the beginning of a study.

A global definition file (.gdf) is the first step towards creating a focal master file (.fmf).

First, click [New] to create a new .gdf or [Open] to modify an existing one.

Behaviours

Use the [Add row] button to add rows. Specify a single alpha-numeric key stroke for each behaviour, define it, and describe it.

Note: JWatcher is case-sensitive: upper and lower case key codes are treated separately. Punctuation key codes are permissible, but the space bar, return, tab, delete and other 'function' keys cannot be defined.

Use the [Delete row] button to remove rows.

Modifiers

Use the [Add row] button to add rows. Specify a single alpha-numeric key stroke for each modifier, define it, and describe it.

Use the [Delete row] button to remove rows.

Type in descriptive supplementary information in the bottom text box.

Use the [Save] button to save your .gdf file when complete.

4 Focal Master

Focal Master creates a set of specifications (the .fmf file) used when recording data using Data Capture. It defines a set of questions the user will be asked prior to capturing the data. The answers to these questions (e.g., identity of focal subject, date, time of day, habitat, etc.) will be added to the header of the resulting data file (.dat), and will also be reported in any results files generated when analysing the data file. The focal master file also contains the duration of a focal observation. It is best used AFTER creating a global definition file, although this is not required. A .fmf file is also a pre-requisite for analysing data as it is required to create a 'focal analysis master file' (.faf) and, if needed, to later create 'combination master files' (.cmf).

4.1 Focal Master > Test Details

Use the [New] button to open a previously created .gdf file and import the existing behaviours and modifiers. Use [Open] to open an already existing .fmf. Clicking either button will take you into a navigation window.

Give the .fmf file a name and click [Open] to return to the Test Details window.

Define the observation session in hours, minutes, and seconds. Note, it is sufficient to fill in only one box (e.g., '1' in the minutes box will create a 60 second focal). If no session duration is specified, data capture will continue until stopped.

Select 'count up' or 'count down' to specify whether you want the timer in Data Capture to be increasing to the time limit, or decreasing towards 0.

Add any notes to the text box.

4.2 Focal Master > Questions

Define up to 6 questions the user will be prompted with prior to capturing data. For example, you might want to note the identity of the focal subject, the time of day, the weather, etc.

4.3 Focal Master > Define Codes

If you have imported a .gdf file, there should be a series of already defined behaviours and modifiers. Using the [Add row] or [Delete row] buttons, edit this list to include just those behaviours and modifiers you are interested in quantifying for your current project.

Use [Save] to save the file with the existing name (the [Save] box is only selectable if you have made any changes) and the [Save As] box to save the file with a new name.

Note: Making changes in the .fmf will not modify the .gdf from which it was created.

5 Data Capture

Data Capture is JWatcher's event-recording program. Use this to record the onset of behavioural states (e.g., the onset of a bout of foraging) or the occurrence of behavioural events (e.g., the occurrence of an alarm call). Data capture logs any and all definable key presses (see section 3), whether or not they are defined in the .fmf file.

5.1 Data Capture > Data File

In this window, you will have to both name your data file and select an .fmf used for data capture.

In data file field, select the path where the data file will be stored by clicking on the file icon. If you do not select a pathway, the data file will be stored in the default location—in the Application folder where JWatcher is installed. Selecting the file icon will open a navigation window. Using the pull-down directory in the navigation window, specify the location where you want to store the data file. Name the data file and click the [Open] box. If you select an existing .dat file you will overwrite it.

In the focal master field, select a focal master file (.fmf) that you want to use while scoring behaviour by clicking on the file icon.

Note: You do not need a .fmf file at this time, but it will facilitate scoring and later analysis if data files are scored using a .fmf file.

When you have named your data file and selected a .fmf file, click [Next].

Note: The first time you open JWatcher after a shut-down it will automatically revert to the default pathway. Once you select a path, JWatcher will save that path (until changed) to facilitate subsequent scoring sessions. However, this path will be lost when JWatcher is closed.

5.2 Data Capture > Questions and Answers

In this window, answer the questions that have been specified in the .fmf file.

When finished, click [Next].

5.3 Data Capture > Capture Data

Using the mouse, click [Start] to begin scoring behaviour. The session will either time out at the end, or you can click [Stop] to end an observation session before the pre-determined time.

Alternatively, you can move the cursor around using the tab key. Ensure that the 'focus' is on one of the duration boxes (i.e., there is a blinking cursor) and tab until the [Start] button is highlighted. Depressing the space bar will now start the scoring process.

If no session duration is specified, the data can be recorded until stopped.

The View [Behaviours] button places a list of behaviours and their key codes into a separate window. Doing so may facilitate observation.

The View [Modifiers] button places a list of modifiers and their key codes into a separate window.

When finished, the file is automatically saved to the location specified. Click [New] to score the next focal session.

5.4 Note on 'initialising' a focal

Some event recorders require the user to initialise behavioural states before scoring behaviour. We elected to allow the user to begin capturing data immediately. Therefore, to initialise a behavioural state, the user has two options:

- 1) Quickly hit a key after starting JWatcher. The time delay between starting JWatcher and hitting the first key may be unimportant if the focal duration is relatively long.
- 2) In cases where the time delay may influence later analyses, use the Edit Data function (see section 6) to edit the data set to begin at time = 0.

6 Edit Data

Edit Data allows you to modify the key strokes and times created in Data Capture.

Use the [Open] button to open a data file (data files have the .dat suffix).

Use the [Insert Row] button to add rows, the [Delete Row] button to delete rows, and the [Save] or [Save As] buttons to save changes.

Notes: Type or edit the time in the 'Elapsed' column; it will automatically update the non-editable hh:mm:ss:ms column. If the .fmf is either in the same directory as the data file, or is in the same location as it was when the data file was scored, then typing a recognized behaviour or modifier key code in the key pressed column will result in the appropriate behaviour or modifier being typed in the Behaviour/*Modifier column. Ensure that new entries are chronological. Non-chronological entries will be treated as errors during analysis and no results files will be produced.

7 Focal Analysis Master

Focal Analysis Master creates a focal analysis master file (.faf) which specifies how data files (.dat) will be analysed and specifies the results to be reported.

JWatcher's 'Score once, analyse many times' philosophy is embedded in the focal analysis master file. A single data file can be analysed differently by changing definitions in a focal analysis master file. Users may wish to modify time bins, subtract or ignore certain behaviours, re-define mutually-exclusive behaviours, modify how incomplete time bins are handled, or explore the differences between state and event analyses.

However, **logical errors when selecting non-default options can generate meaningless results.** Be VERY careful setting up focal analysis master files. We suggest paying particular attention to the logical relationships specified.

Use [New] button to select a focal master file (.fmf file) that contains the key codes to be analysed.

Use [Open] button to open a pre-existing focal analysis master file (.faf file).

Note: Ensure that you use the the same .fmf to score behaviour and to analyse behaviour. The analysis routine will later check whether the same .fmf was used to capture data and to create the .faf for analysis. If not, JWatcher will inform you that different .fmf files were used. However, analysis will proceed even if the .fmf files do not match.

7.1 Focal Analysis Master > Modifiers

Specify behaviours that may have modifiers following them. For instance, aggression ('a') may be directed to different individuals (e.g., '1', '2', or '3'). Default is no modifiers follow any behaviours.

Note: Only modifier key codes that immediately follow a behaviour specified as having modifiers will be reported in the results files. Any modifier key codes that do not follow a modified behaviour will be recorded in the .dat file, but will be ignored during analysis.

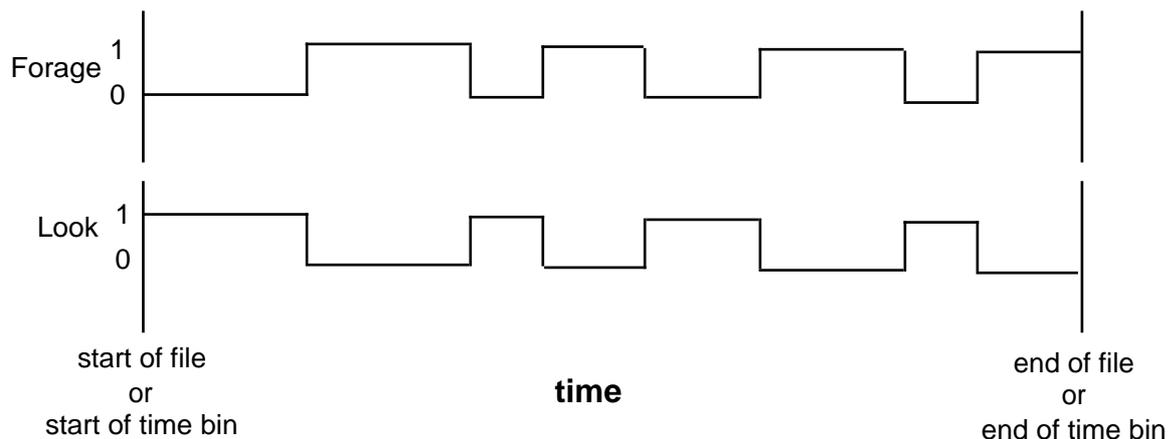
7.2 Focal Analysis Master > States

States are behaviours that occur over time. For instance, a subject may forage for 2 minutes. The onset of the bout of foraging could be noted by hitting 'f' in data capture. Some behaviours may be mutually-incompatible with foraging (e.g., l-locomotion), whereas others may not be (e.g., c-chewing). Mutually incompatible behaviours signal the termination of a state and are used to calculate durations of states. Thus, a bout of foraging begins when 'f' is hit and ends when a mutually-incompatible behaviour key

code is hit—in this case 'l'. This window allows the user to specify precisely which behaviours are mutually-exclusive with each other.

The default is that all behaviours are mutually exclusive; the onset of one behaviour turns off another behaviour (Figure 2). De-selecting the 'all codes are mutually exclusive box' allows the matrix to be modified. However, codes are always mutually exclusive with themselves.

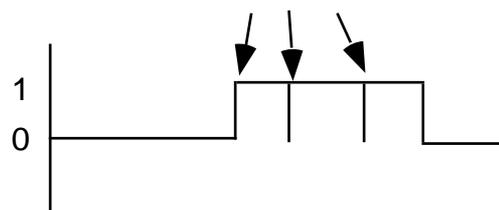
Figure 2. Traces of two mutually-exclusive behaviours.



It is possible, and in some cases desirable, to hit the same key successively (Figure 3). For instance, if you are interested in studying bouts of vigilance behaviour, a bout begins every time a subject moves its head. In this case, because codes are mutually exclusive with themselves, the duration of each bout is measured as the time in between each key stroke. The interval between bouts is 0. The number of events of vigilance is the number (N) of keystrokes. The average duration of each bout of vigilance is the total duration divided by N-1 keystrokes.

Figure 3. Details about event and state analyses.

In State Analysis, repeated key strokes
lead to intervals with no duration



7.3 Focal Analysis Master > Exclusions

The 'subtract' function subtracts the total time for a specified state from the total duration of a time bin when calculating time budgets. For example, you might wish to calculate the time allocated to running as a function of total locomotion. In this case behaviours other than those involved in locomotion could be subtracted. The proportion of time and proportion of time in sight statistics reported for each behaviour in the results will be adjusted accordingly.

Subtract only influences duration statistics, but not interval statistics.

Subtract does not affect the logical relationships with other behaviours. Time is removed, but subtracted behaviours still terminate bouts of mutually-exclusive behaviours.

Note: Subtracting a behaviour will generate different results than specifying that same behaviour code as representing 'out-of-sight' (see 7.5 for more information on out-of-sight and its effects). Why? Scoring out-of-sight means that you are unsure of what is going on. This has consequences for both interval and duration statistics. In contrast, when you subtract a behaviour you know what has occurred; subtraction simply removes that time from the denominator when calculating time budgets.

'Ignore' literally ignores selected behaviours when calculating states. Ignored behaviours, therefore, do not turn off mutually exclusive behaviours; it is as if the key code was never pressed.

Note: This can potentially cause nonsensical output if some behaviours in a related set are ignored, while others are retained for analysis.

Ignore has precedence over subtract. Defaults are no behaviours are either subtracted or ignored. Ignored behaviours are not reported in the results, while subtracted behaviours are. For subtracted behaviours only, the proportion of time and the proportion of time in sight statistics are calculated as the total time for the subtracted behaviour divided by the total duration of the time bin. The total time for the subtracted behaviour is not subtracted from the denominator in this case.

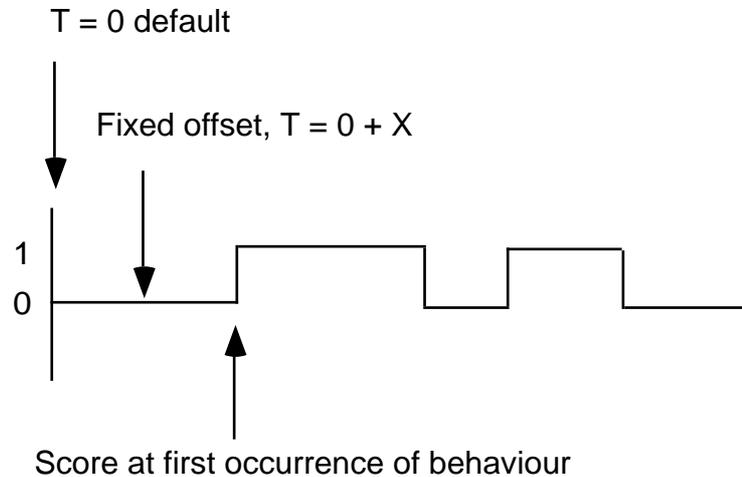
Warning: If you subtract a behaviour that has been designated as 'out-of-sight' (see 7.5), the proportion of time in sight statistic is meaningless.

7.4 Focal Analysis Master > Time

Decisions made in this window influence the time over which behaviour is analysed. The default setting is for the entire data file (.dat file) to be analysed. However, there are cases where users may wish to modify this in several ways.

Users may specify at what point to begin analysing a file (Figure 4). De-selecting the 'Score from beginning of session' box allows modifications.

Figure 4. Score from beginning variations.



Fixed offset specifies the number of seconds from the beginning of the data file that should be ignored in analysis. Users may wish to use a fixed offset to get around the problem of the time lag between when the start button is pressed and the first key stroke. Note, however, ALL key codes hit before this fixed offset will be ignored.

First occurrence defines which, if any, behaviour will trigger the onset of analysis. When a specific code has been designated, all key codes hit before this code will be ignored. You may also also analyse from the onset of any keycode, regardless of its identity. Choose the <any> option from the pull down menu. This is another potential solution to the time lag problem discussed above.

Note: Using the <any> option will begin analysis at the first recorded key stroke regardless of whether it is defined in your .fmf, and regardless of whether it has been 'ignored'. In the latter case, a warning will be given during analysis, but analysis will proceed.

Time bin duration specifies how the data file (.dat file) is subdivided for analysis. For instance, a user may wish to score a 5-minute focal sample but summarise behaviour in 1-minute blocks of time. Specifying the time bin duration in 60 second blocks will take a 5-minute focal and generate 5 different 60-second blocks of results.

Note: '0' time bin duration is the default and specifies that the entire focal will be analysed in one time bin.

Various combinations of time bin duration and offset times may lead to the case where the last time bin is truncated (i.e., it is shorter than previous time

bins). Choose 'End with last complete time bin' to only report those results for complete time bins or 'Report all' to report those results for all time bins.

7.5 Focal Analysis Master > State Analysis

For each behavioural state, JWatcher calculates the number of bouts, the total time summed over all bouts, the average bout duration, the standard deviation of bout duration, and the proportion of time a focal subject engages in a given behaviour calculated over the entire time bin.

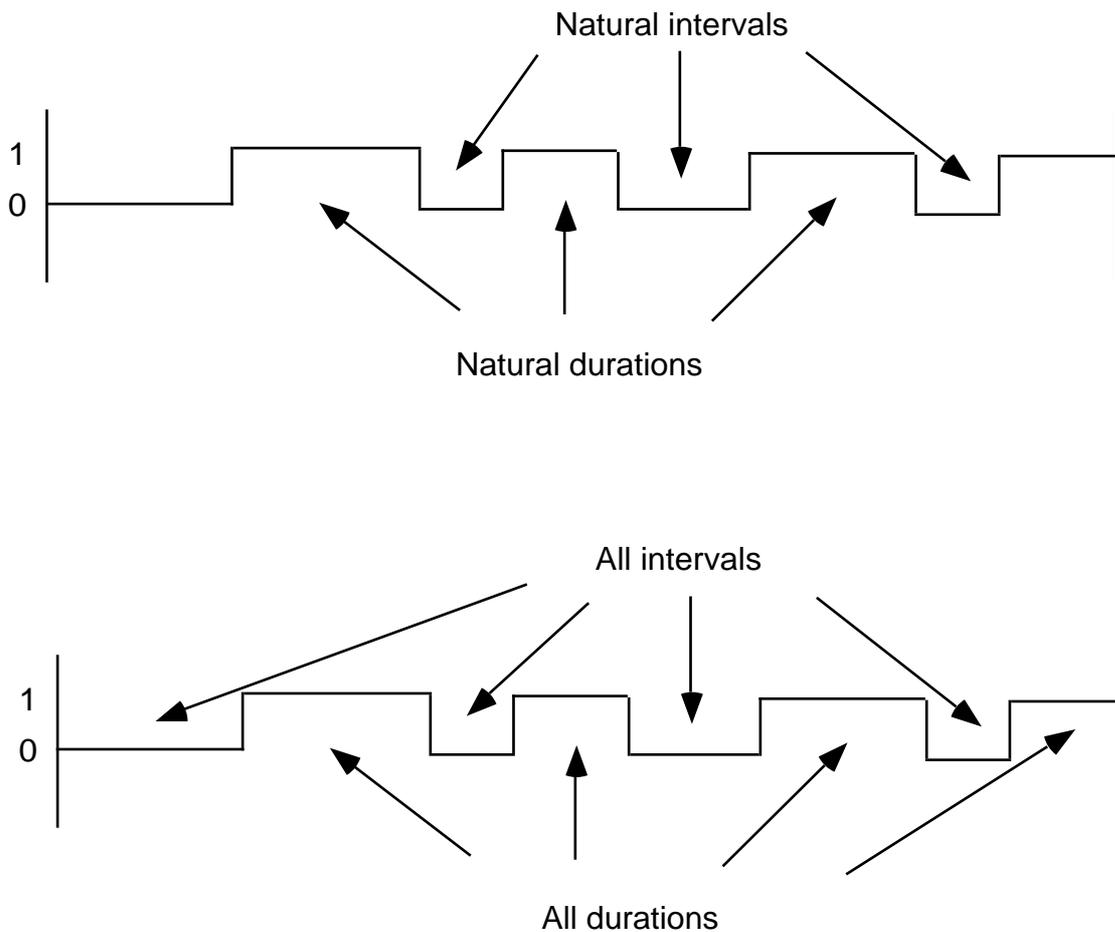
Note: The total observation period is derived from EOF ('end of focal') time in each .dat file and not the focal duration specified in the .fmf.

Subjects are not always in view. Users may wish to specify a key code used when the subject is out-of-sight (e.g., 'o'). The proportion of time allocated to each behaviour as a function of time in sight can also be calculated.

A behavioural state has a duration, and there is an interval of time between bouts of behaviour. These durations and intervals may be either 'natural' (i.e., they start and end within a time bin), or they may be truncated by either a time bin border or the end of file (Figure 5). Selecting 'natural' interval and 'natural' duration reports statistics only on those intervals and state durations not truncated. Selecting 'all' interval and 'all' duration reports statistics on all occurrences of interval and state durations regardless of whether they are truncated.

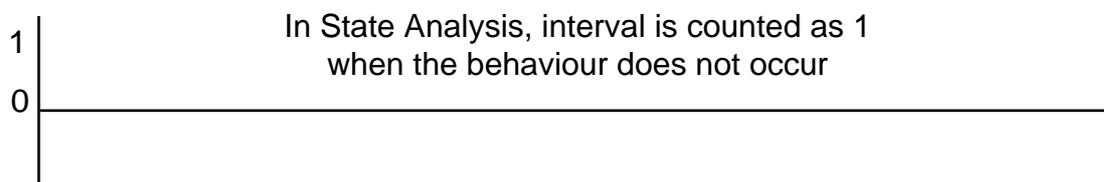
Note: The onset of out-of-sight truncates intervals and durations. Scoring out-of-sight implies that you are unsure as to what behaviour(s) focal subject is engaged. If you are interested in natural durations or intervals, any bout of behaviour adjacent to a bout of out-of-sight is, by definition, truncated.

Figure 5. Interval and duration definitions.



If a behaviour does not occur during a time bin, the interval is considered to occur once (i.e., $N = 1$), with a total duration equal to the time bin (Figure 6).

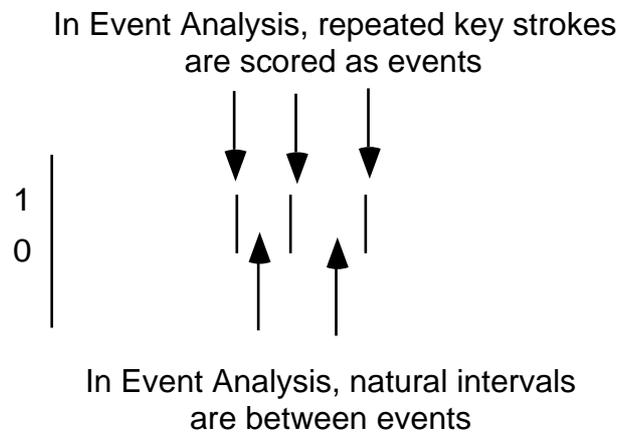
Figure 6. Details about State Analysis.



7.6 Focal Analysis Master > Event Analysis

Events are behaviours that have no duration. Thus, they can be counted and intervals between them quantified. For event analysis, JWatcher only quantifies 'natural' intervals (see section 7.5 for more about 'natural' versus 'all' intervals).

Figure 7. Details about event analysis.



Note: JWatcher permits behaviours to be analysed as both events and states.

Select those behaviours for which event analysis is requested and select the output required. The first check box selects both 'event count' and 'event natural interval count'. Event count is a tally of the number of times an event occurred, while the natural interval count is the number of natural intervals between events. 'Average interval between' and 'Standard deviation' are based on natural intervals. Remember that if you specified an out-of-sight key in the State Analysis window, out-of-sight will truncate all natural intervals including those in event analysis.

Once you have selected analysis options, save changes using [Save], or create a new file name using [Save As].

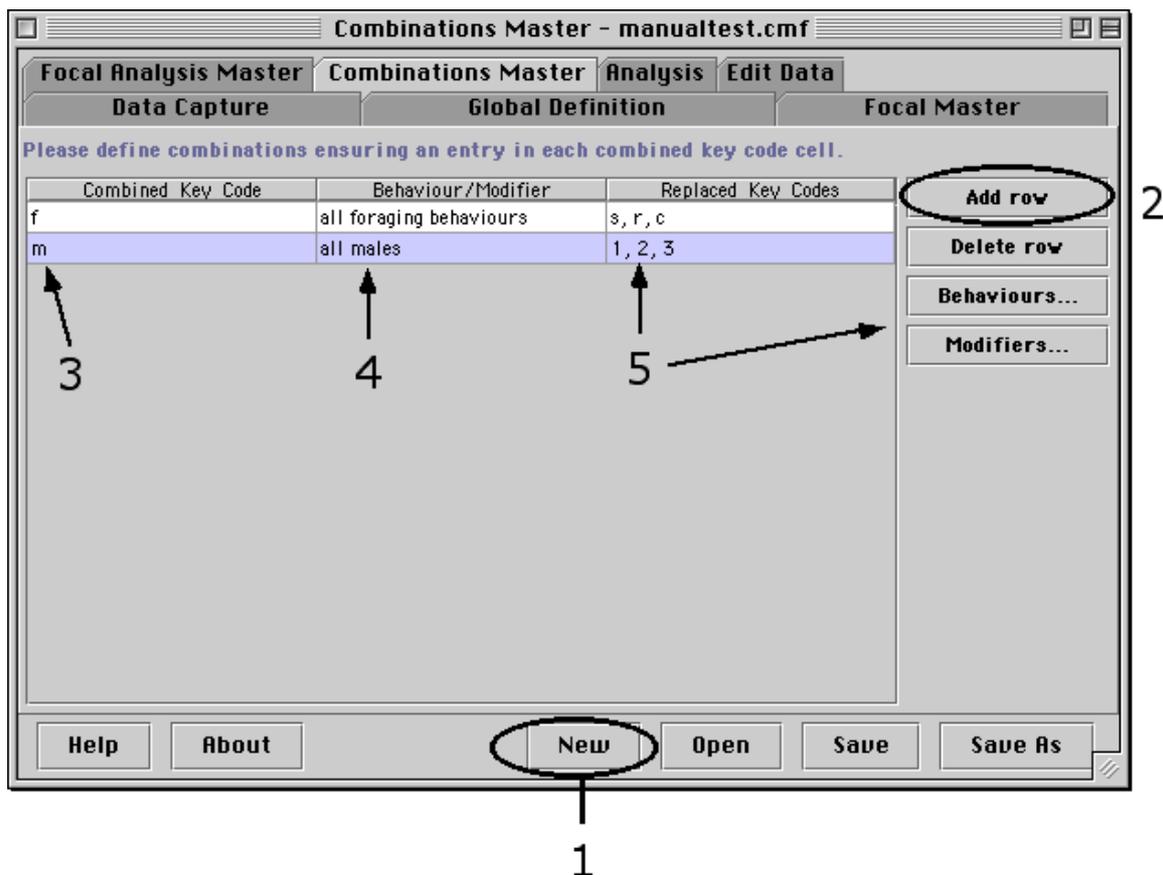
[Summary] opens a text window containing all the specifications in the current focal analysis master file (.faf). Select the text using a mouse, and 'control+c' to copy the contents of the text window. Contents can now be pasted into a word processor and printed.

Note: The section under the heading 'warnings' in the summary will list potential problems that may arise during analysis.

8 Combinations

Combinations Master allows you to combine key codes in data files (.dat files) and to ultimately create a new 'combined' data file. To do so, you must first create a combinations master file (.cmf) which is later used by the Analysis > Combine Key Codes routine.

Note: You may want to have a print-out of the .fmf file you are modifying next to you when creating a .cmf file. To print it, open the file in a word processor or text editor. Alternatively, you could view your .fmf by opening it in the Focal Master window and tabbing between windows.



- 1) Click the [New] button and, using the navigator box, select a pre-existing focal master file (.fmf). This inherits a set of key codes that can now be combined.
- 2) Use the [Add row] button to add a row.
- 3) Type a key code that will replace one or more existing key codes. For example, f--forage may be used to replace a series of scored foraging behaviours (e.g., s--stand and forage, r--rear and forage, c---crouch and forage).

- 4) Define the newly created key code in the Behaviour/Modifier text box.
- 5) Click on the 'Replaced Key Codes' text box and either type the key codes (separated by a comma) to be replaced, or click on the now highlighted [Behaviours...] button to see a list of the key codes. Use the control key to select multiple key codes to be combined. Clicking OK places those key codes into the 'Replaced Key Codes' text field.

Note: You can replace or re-define behaviours or modifiers, but not a mix of behaviours and modifiers.

Combine or re-define additional key codes using the above procedure.

When finished, save the newly created .cmf file by clicking on the [Save] box and defining the new file name in the navigator window.

The next step is to combine data files. This is done in the Analysis window under the Combine Key Codes tab (see section 9.2).

Note: When your newly created .cmf file is saved, a new .fmf file with the replaced key codes is automatically created and placed in the same folder as your original .fmf. By default, it is named c*.fmf, where * = the name you gave the .cmf. Before analysing combined files, you will have to create a new .faf which is based on the c*.fmf. To create a .faf, see section 7.

9 Analysis

In the Analysis tab, you can either run your .faf to analyse .dat files, or run your .cmf to combine key codes in .dat files. Both routines process either single data files or work in a batch-processing mode.

9.1 Analysis > Analyse Data

You must have a focal analysis master file (.faf file) and at least one data file (.dat file) to run Analyse Data.

Click on the file icon to specify the .faf file used to analyse the data.

Click on either the file icon or the folder icon to specify a .dat file or a folder with one or more .dat files to be analysed. All .dat files in a folder will be analysed if the folder option is selected.

Use the folder icon to specify a folder in which the results file(s) will later be placed. JWatcher will create a new folder if one does not already exist.

Note: JWatcher names the results files by changing the suffix of the original .dat file name. For example, the results file generated by analysing a data file named 'test.dat' will be 'test.cd.res' and 'test.tr.res' (see below for the distinction between *.cd.res and *.tr.res). If you modify the .faf file and re-run an analysis, be sure to specify a different folder for the results; the default is to write over existing results files. JWatcher will ask if it is OK to overwrite files before doing so.

There are two types of results files output; both are comma delimited text files.

*.cd.res is a 'codes down' results file suitable for importing into a spreadsheet or word processing program. After a specified header, each row of results summarises a behaviour or a behaviour and modifier. Each column in the results file contains a summary statistic specified when creating the focal analysis master file (.faf file).

*.tr.res provides an easy way to graph 'traces' of behaviour over time using a spreadsheet or graphics program. All 0-1 and 1-0 'transitions' along with the transition times are specified for each behaviour. If behaviours have been subtracted, 0-'-1' transitions illustrate where subtracted behaviours have occurred in a time window.

Checking the box 'Include summary of the focal analysis master in the results file' will do just that.

Decide whether to report the results for all behaviours and modifiers in the focal analysis master file, or only those which occurred in a data file by selecting the appropriate radio button.

Select the [Analyse] button to run the program. A progress box is updated each time a data file is analysed. A text box provides an option to examine a file which contains a list of any errors or warnings. If an 'error' occurs, no results files will be produced. However, in the case of a 'warning', the .dat file will be processed. This log file (.log file) is saved into the active directory with a unique date/time-based name.

Select the [Reset] button to clear the text fields, but note that the pathways and file names are editable.

Select the [View Log] button to view the log file created when analysing data.

To view results files, open them in a word processing or spreadsheet program.

9.2 Analysis > Combine Key Codes

You must have a combinations master file (.cmf file) and at least one data file (.dat file) to run Combine Key Codes.

Click on the file icon to specify the .cmf file to be used to combine/replace key codes within the data file.

Use either the file icon or the folder icon to specify a .dat file or a folder with one or more .dat files to be combined. All .dat files in a folder will be combined if the folder option is selected.

Type in a prefix that will be used when naming the 'combined' files (e.g., typing 'c' will rename a data file named 'test.dat' to 'ctest.dat').

Use the folder icon to specify a folder where the combined file(s) will be placed; JWatcher will create a new one if one does not already exist.

Note: JWatcher names the combined files by adding the specified prefix to the original .dat file name. If you modify the .cmf file and re-run Combine Key Codes, be sure to specify a different folder for the results; the default is to write over existing combined files. JWatcher will ask if it is OK to overwrite files before doing so.

Select the [Combine] button to combine key codes.

Select the [Reset] button to clear the text fields but note that pathways and file names are editable.

Select the [View Log] button to view the log file created when combining key codes.

To analyse these files you must first create a new .faf based on the c*.fmf. See section 7.

10 Strategies for using JWatcher

To use JWatcher, you must first create an overall ethogram for your species in the .gdf. Then create a .fmf from a subset of those behaviours. Once you have a .fmf, you're ready to score behaviour. However, observing behaviour ultimately requires you to have an explicit idea of what you're aiming to quantify.

We suggest that you create an .faf before scoring behaviour as the process will force you to decide whether or not behaviours are mutually-exclusive, whether or not you will ultimately want to subtract or ignore behaviours, which behaviours may or may not have modifiers, and whether or not you're interested in understanding something about events or states. Considering these issues before scoring behaviour does not prevent you from changing the .faf at a later time. In fact, we find the ability to 'time window' data one of the most useful functions of JWatcher and we routinely analyse data with several .fafs.

For state analyses, it behoves you to think very clearly about the difference in meaning between 'natural interval' and 'natural duration' results compared to 'all interval' and 'all duration' results (Figure 5). In some circumstances, you can easily generate nonsensical, but logically correct, answers. For instance, if out-of-sight is mutually exclusive with all other behaviours (as it should typically be), it will truncate an interval causing the 'all interval occurrence' count to increase by 1 each time an out-of-sight occurs. Since you have no idea what happened when a subject was out-of-sight, and because interval statistics typically make more sense when they are based on natural intervals, the 'all interval occurrence' answer, while logically correct, isn't that meaningful.

There are two ways one could conceivably use JWatcher to calculate the proportion of time in sight. One way is to 'subtract' the behaviour that represents out-of-sight by using the subtract function in Focal Analysis Master > Exclusions and then examine the proportion of time statistic. The other way is to specify an out-of-sight key in Focal Analysis Master > State Analysis and examine the 'proportion of time in sight' statistic. However, designating a key as out-of-sight in the State Analysis window will have additional consequences beyond simple subtraction. Specifically, it truncates natural durations and any intervals. If a focal subject is truly out of sight, this is a desirable characteristic.

When .dat, .faf, and .cmf files are created, the location (i.e., pathway) to the associated .fmf at the time of creation is recorded. Editing .dat files, or using .faf or .cmf files for later analysis, requires the .fmf either to be in the original location where it was when the above files were created, or in the same folder as the .dat, .faf, or .cmf files. We suggest that users create a directory for each project and not move around the .fmf, .faf, or .cmf files. Data and results files can usually be reorganised as long as the .fmf stays in its original location.

11 Warnings and Errors

Analysis

Warnings:

"The focal master for focal data is <filename1>. This does not match the focal master <filename2> in focal analysis master. Analysing anyway."

"Modifier [<m>] at offset [<offset>] does not follow a behaviour. Ignoring modifier."

"Modifier [<m>] at offset [<offset>] follows an already modified behaviour [<m>]. Ignoring modifier."

"Code [] at offset [<offset>] not in focal master <filename>. Ignoring code."

"Length of time bin [<time bin duration>] is greater than length of focal data capture session [<duration>]."

"Scoring from first code [<code>] but this behaviour is ignored."

Errors:

"There are no code entries in the focal data file. Skipping analysis."

"Unable to find focal data file <filename>"

"Unable to find related focal master: <filepath>
To proceed, locate focal master and place in: <directorypath>"

"Behaviours in the related focal master <name> have changed.
Edit the focal analysis master to update."

"No results have been selected for reporting in this focal analysis master"

"Behaviour to score from [] not found in focal data. Skipping analysis."

"Offset [<offset>] matches or exceeds the duration of the data capture session [<duration>]. Skipping analysis."

"Scoring from first occurrence of code [<code>] but this behaviour is ignored. Skipping analysis."

"No complete time bins to report on." (End with last complete bin)

"No complete or truncated time bins to report on."

"The number of time bins calculated is [<number>]. Greater than 1000000 is too many to attempt to score!"

"Code [<c>] at offset [<offset>] is earlier than previous entry. Skipping analysis."

"Code [<c>] at offset [<offset>] is later than focal data EOF time. Skipping analysis."

"Unable to save results to <filename>."

(Any of the load exceptions below may also be reported as errors).

Combining Key Codes

Warnings:

"The focal data file does not contain any key codes to combine."

"The focal master for focal data is <filename1>. This does not match the focal master <filename2> in combinations master. Combining anyway."

Errors:

"Unable to find focal data file <filename>"

"Unable to save combined data to <filename>."

(Any of the load exceptions below may also be reported as errors).

Load exceptions

"This <filetype> contains no information: <filepath>"

"Unable to find <filetype> <filepath>"

"Unable to load <filetype> <filepath>"

"Unable to close <filetype> <filepath>"

"Unable to load from the format: <fileformat>"

Save exceptions

"Unable to save to file <filepath>"

"Unable to save incrementally in the format: <fileformat>" (Focal Data capture only)

"Unable to save in the format: <fileformat>"

Other

When opening focal analysis master to edit:

"Unable to find related focal master: <filepath>

To proceed, locate focal master and place in: <directorypath>"

"Behaviours in the related focal master file <filepath>
have changed. OK to update Focal Analysis Master?"

When opening Combinations Master File or Focal Data File for editing:

"Unable to find related focal master: <filename>

You may edit but cannot view behaviours and modifiers."

When creating a new focal master file:

"Please select an existing Global Definition File."

When creating a new focal analysis master file:

"Please select an existing Focal Master File."

When creating a new combinations master file:

"Please select an existing Focal Master File."

When setting up to capture focal data and a focal master file has been entered:

"<focalmasterfilepath> does not exist."

When preparing to analyse focal data or combine key codes:

When checking Output Folder:

<path> is not a Folder."

When checking Focal Data Files before processing:

"<filepath> is not a focal data file or a folder"

"No focal data files have been found"

When data has been analysed or combined:

"Unable to display log file <filepath>"

12 Frequently asked questions about JWatcher

Last updated 14 November 2000

Q: What is JWatcher?

A: JWatcher is an event-recording program written in Java, a computer language developed by Sun Microsystems that runs in both the Windows and Macintosh (and UNIX and LINUX...) operating environments.

Q: What is an event recorder?

A: Event recorders are tools to help quantify behaviour. By logging key presses that signify the onset or occurrence of behaviours, event recorders allow you to estimate the time animals or people allocate to different activities. To learn more about quantifying behaviour, read the outstanding book *Observing Behaviour* (now in its second edition) by Paul Martin and Pat Bateson (Cambridge University Press).

Q: How much does JWatcher cost?

A: JWatcher is free to use. Please tell others about it and use it in your laboratories and classes. To ensure you're using the correct version, download it directly from the JWatcher page (<http://galliform.psy.mq.edu.au/jwatcher/>).

Q: Who developed JWatcher?

A: JWatcher was developed by Dan Blumstein, Chris Evans and Janice Daniel at the Animal Behaviour Laboratory at Macquarie University in Sydney, Australia. First stage development was supported by a grant from Macquarie, and coding was done by Fiona Walkerden, Xuhong Li and Derek Renouf of Adaptive Arts Pty Ltd (<http://www.adaptive-arts.com/>), a company that develops data mining and database management tools in Java. While we are interested in bug reports (email to: jwatcher@galliform.psy.mq.edu.au), and we will update this FAQ periodically, we are currently unable to correspond with users about the routine workings of JWatcher. JWatcher comes with no warranties expressed or implied.

Q: What operating system does JWatcher require?

A: JWatcher was developed in a Windows (98/NT) environment and tested extensively using both Windows (95/98/NT) and the Mac OS (System 8.5, 8.6 and 9.0). It should run on any computer that can run the Java Virtual Machine (also called the Apple Applet Runner in the Mac OS). It was developed to work on the Windows Java Runtime Environment 1.1.8 and the Mac OS Runtime For Java version 2.2.3. Needless to say, it runs faster and more smoothly on newer machines with a lot of RAM.

Q: I can score behaviour but JWatcher freezes when I try to analyse it. What's happening?

A: You should use JWatcher's analysis routines on a computer with more RAM. Data Capture works well with limited RAM but analysing or combining keycodes work better with more RAM.

Q: How do I open/print a results file?

A: Results files (.res) are comma delimited text files. They can be opened within any word processing or spreadsheet programs.

Q: JWatcher does not redraw the screen properly and suddenly begins behaving badly. Why?

A: You may have too many applications open. Quitting JWatcher and closing un-used applications may solve this problem.

Q: How can I print my .faf to verify what it does?

A: In Focal Analysis Master, click on the [Summary] button, select all of the contents of the summary file using a mouse, and then copy (control + c) and paste the contents into a word processor.

Q: Why do I have to specify the path where I want files to be stored every time I restart JWatcher?

A: JWatcher does not create a preference file. Thus, the default location where files are created and stored is the folder where the application resides. When we use JWatcher, we store the application in the default installation location (in the applications folder), and create a folder elsewhere where we store JWatcher files. Each time we restart Jwatcher, we have to navigate over to that folder. Within a session, however, JWatcher remembers the last location used.

Q: How can I print my data files?

A: Data files (.dat) are comma delimited text files. They can be opened within any word processing or spreadsheet programs.

Q: How do I edit a pre-existing data file?

A: Use the Edit data function built into JWatcher, or edit using a word processor or spreadsheet program.

Q: If I analyse the same data file using first one and then another focal analysis master file, my results file is overwritten. How can I avoid this?

A: Select another destination folder into which the second batch of results can be put. By design, results are named by adding either cd.res or tr.res to an existing .dat file.

Q: When I re-size the JWatcher screen, some of the buttons disappear. Why?

A: JWatcher is designed to run on a 640 x 480 screen. Re-sizing it below that may cause some problems that should be easily reversed by making it bigger.

Q: My on-line help doesn't work when I'm running JWatcher.

A: We've had some documented problems with Netscape Navigator™ running on a Macintosh recognising our help files. Try downloading Microsoft's Internet Explorer and setting it as your default web browser. Additionally, depending on your memory allocations, JWatcher may not be able to run properly with a web browser. Print the JWatcher guide directly from the .pdf file provided and use it, rather than the on-line help.

Q: How much memory is needed to run JWatcher?

A: JWatcher only requires about 2 mb of RAM. However, the Java Virtual Machine requires more and takes it. For best performance, we suggest having about 64 mb of RAM free when running JWatcher. Allocating different amounts of RAM to JWatcher and the Java Virtual Machine may result in strange performance.

Note: Strange performance in no way influences the results, but you may have to quit the program and restart it periodically.

Q: I now have a bunch of JWatcher .res files, but I still don't know how much time my animals allocate to behaviour. What should I do?

A: JWatcher creates results files for each focal animal sample. Presumably your time budgets will be based on multiple focal animal samples conducted on multiple individuals. You will have to combine, using a spreadsheet, your JWatcher results files and calculate an overall time budget.

Q: What is the .tr.res file?

A: The .tr.res file is a results file set up in a way that you can easily graph a 'trace' of behaviour. Behavioural traces show the onset and termination of bouts of behaviour.

Q: When I want to quantify a behaviour and treat it as both an event and a state, I hit the same behaviour code more than once in a row. However, in state analysis, each key entry is counted as a bout with a 0 second interval between them. Is this correct?

A: Yes. Logically, each behaviour turns itself off. Therefore, you must be careful to decide whether you want the same behaviour code to be used to signify events within states or to solve this problem by assigning different codes to signify the state and event.

Q: I don't want to have to remove my hands from the keyboard to start recording behaviour in Data Capture. Is there a way I don't have to use the mouse to click start a focal observation?

A: Yes. Using the tab key, move the focus to the start button. When the focus is on the start button it will be highlighted. Once highlighted, hitting the space bar will start the clock.

Q: I'm working with a lot of slightly different .faf files. How can I keep track of them?

A: Data management is always difficult when conducting complex analyses. Copy the contents of the .faf summary into a word processor, add relevant comments, and print and save it.

Q: How can I view my .fmf file?

A: Open it and print it from a word processor.

Q: Why do I get unrecognised key codes in the analysis log file?

A: Either you have mistakenly typed an unrecognised key code or the .fmf used to score data is different from that used to create the .faf.

Q: How come I can't save my .faf?

A: Ahh, that nasty RAM again. Close other programs and/or try restarting JWatcher.

Q: I opened my .dat file to edit it and received a message saying that the associated behaviours and modifiers are missing. What's up?

A: In order to edit data files, JWatcher must locate and load the .fmf used when scoring data. The .fmf must either be in the location it was when scoring data, or it can be in the same folder as the .dat file.

Q: I batch processed a number of files but in some cases no results files are produced. Why?

A: Check the .log files for errors. Files with serious errors will not be processed.

Q: I was given the error message: "Behaviours in the related focal master <name> have changed. Edit the focal analysis master to update." What do I do?

A: Reopen the .faf in the Focal Analysis Master tab. A dialog box will ask if it is OK to update your .faf file. Click OK to update.