

# Event Analyzer<sup>tm</sup>

Event Detection and  
Analysis System

## User Guide

Version 5.10  
October, 2018

Engineering Design

This document is provided for the sole purpose of operating the **Event Analyzer™**. No part of this document may be reproduced, transmitted, or stored by any means, electronic or mechanical. It is prohibited to alter, modify, or adapt the software or documentation, including translating, decompiling, disassembling, or creating derivative works. This document contains proprietary information which is protected by copyright. All rights are reserved.

ENGINEERING DESIGN MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THE MATERIAL CONTAINED HEREIN, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Engineering Design shall not under any conditions be liable for errors contained herein or for incidental or consequential damages arising from the furnishing, performance, or use of this material.

The information in this document is subject to change without notice.

© 2000-2018 Engineering Design, Berkeley, CA. All rights reserved.

**SIGNAL, Real-Time Spectrogram, RTS, Event Detector, Event Analyzer, Experiment Maker, CBDisk, DartDisk, DTDisk, NIDisk, WaveDisk** are trademarks of Engineering Design.

The following are service marks, trademarks, and/or registered trademarks of the respective companies:

Communication Automation: Dart

Creative Technology: Audigy, Extigy

Data Translation: Open Layers

Hewlett-Packard: HP, LaserJet, and DeskJet

Measurement Computing Corp: Computer Boards

Microsoft: Windows, Windows 95, Windows 98, Windows 2000, Windows XP

National Instruments: NI-DAQ

Engineering Design  
262 Grizzly Peak Blvd  
Berkeley, CA 94708 USA  
Tel/fax 510-524-4476  
Email [info@engdes.com](mailto:info@engdes.com)  
**[www.engdes.com](http://www.engdes.com)**

# LICENSE AGREEMENT

**THIS IS A LEGAL AGREEMENT BETWEEN ENGINEERING DESIGN AND THE BUYER. BY OPERATING THIS SOFTWARE, THE BUYER ACCEPTS THE TERMS OF THIS AGREEMENT.**

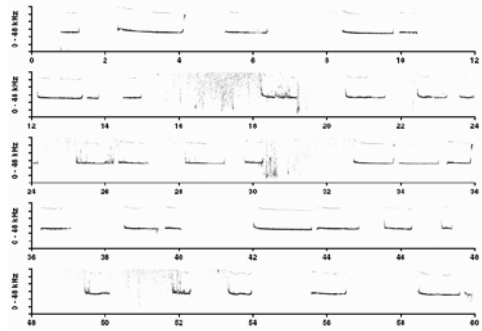
1. Engineering Design (the "Vendor") grants to the Buyer a non-exclusive license to operate the provided software (the "Software") on ONE computer system at a time. The Software may NOT reside simultaneously on more than one computing machine.
2. The Software is the exclusive property of the Vendor. The Software and all documentation are copyright Engineering Design, all rights reserved. The Software may be duplicated ONLY for archival back-up.
3. The Software is warranted to perform substantially in accordance with the operating literature for a period of 30 days from the date of shipment.
4. EXCEPT AS SET FORTH IN THE EXPRESS WARRANTY ABOVE, THE SOFTWARE IS PROVIDED WITH NO OTHER WARRANTIES, EXPRESS OR IMPLIED. THE VENDOR EXCLUDES ALL IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.
5. The Vendor's entire liability and the Buyer's exclusive remedy shall be, at the Vendor's SOLE DISCRETION, either (1) return of the Software and refund of purchase price or (2) repair or replacement of the Software.
6. THE VENDOR WILL NOT BE LIABLE FOR ANY SPECIAL, INDIRECT, OR CONSEQUENTIAL DAMAGES HEREUNDER, INCLUDING, BUT NOT LIMITED TO, LOSS OF PROFITS, LOSS OF USE, OR LOSS OF DATA OR INFORMATION OF ANY KIND, ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE. IN NO EVENT SHALL THE VENDOR BE LIABLE FOR ANY AMOUNT IN EXCESS OF THE PURCHASE PRICE.
7. This agreement is the complete and exclusive agreement between the Vendor and the Buyer concerning the Software.



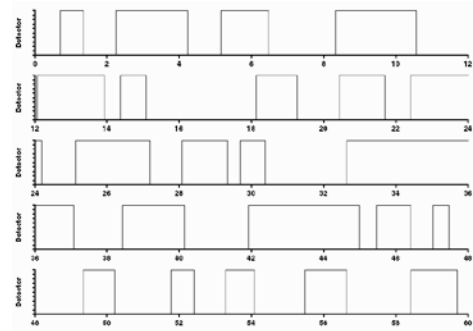
# Table of Contents

1. Overview .....	1
2. Software Installation.....	3
3. Launching the Event Analyzer.....	4
4. Event Detection (EVDX) Overview .....	5
5. EVDX: Menu Overview .....	5
6. EVDX: Program Settings.....	8
7. EVDX: Event Detection.....	14
8. EVDX: Manual Detection .....	16
9. EVDX: Detection Model .....	20
10. EVDX: Detection Function .....	25
11. EVDX: Sound File / Detection Function .....	28
12. EVDX: Event Log File .....	32
13. EVDX: Tutorial .....	35
14. Event Analysis (EVAX) Overview .....	45
15. EVAX: Program Settings.....	46
16. EVAX: Output Displays .....	50
17. EVAX: Tutorial .....	52
18. Operating Hints.....	57





Sound file

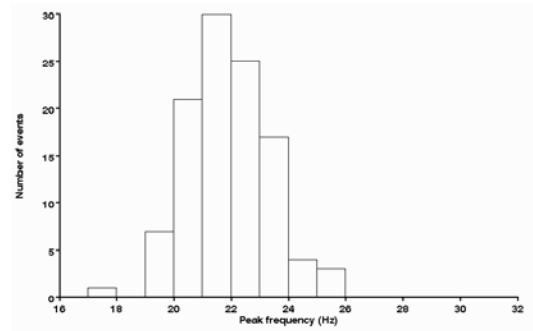


Detected sound events



V2.0 EVANAL event log file							
REC#	TSTART	TINTER	DUR	RMS	FPEAK	FMIN	FMAX
1	0.703	0.703	0.648	0.295	21038	20703	21289
2	2.241	0.890	1.979	0.316	20971	20507	24609
3	5.145	0.925	1.315	0.250	20751	20312	21484
4	8.326	1.867	2.221	0.301	19982	19726	22265
5	12.064	1.517	1.854	0.304	20611	20117	21875
6	14.379	0.461	0.689	0.265	20928	20703	21289
7	18.139	3.071	1.123	0.311	21429	20703	25195
8	20.420	1.158	1.260	0.295	21301	21093	23046
9	22.396	0.716	1.802	0.230	21850	20703	24218
10	25.116	0.918	2.071	0.293	21051	20898	22851
11	28.070	0.882	1.261	0.284	21282	20703	22265

Measured parameters of sound events



Distribution of one parameter

## 1. Overview

The **SIGNAL Event Analyzer<sup>tm</sup>** represents a breakthrough in the analysis of acoustic events as diverse as field recordings of birdsong, laboratory ultrasonic rodent vocalizations, and industrial monitoring of rotating machinery. It can automatically transform long acoustic recordings to summary behavioral acoustic measurements and statistics, such as call rate, call duration, intercall interval, peak frequency, frequency range, etc. Compared to manual event recognition and measurement, the Event Analyzer allows the researcher to investigate many more acoustic parameters over a much longer time period.

The Event Analyzer begins with a continuous digitized sound file containing the input audio stream, automatically detects all events of a specified character, measures a suite of event parameters (including time of occurrence, event duration, principal event frequency, and FM frequency range), stores these parameters in a data file, then produces summary statistics and histogram displays for a variety of parameters over the entire data set. Input sound file duration is virtually unlimited, so the analysis period can be very long.

The Event Analyzer automates and integrates the separate processes of event detection, event measurement, and statistical analysis and display. It reduces these processes to a collection of menus, configuration settings, automatic and interactive measurements, diagnostic tools, statistical analyses, and histogram displays, saving the user potentially months of custom programming. The Event Analyzer utilizes the SIGNAL Event Detector and a variety of

SIGNAL analysis tools, and so both SIGNAL and the SIGNAL Event Detector are required in order to use the Event Analyzer.

Principal features of the Event Analyzer include:

- Automate the event detection process
- Organize detection and analysis settings into configuration menus
- Provide diagnostic tools for visualizing and refining detection
- Accept source sound files in SIGNAL, Wave, or AIFF format
- Optionally save detected events as individual sound files for examination and measurement
- Perform event measurements such as RMS level, peak frequency, and inter-call interval
- Provide both automatic and interactive modes for detection and measurement
- Collect event measurements in a data file for statistical analysis
- Perform statistical analyses such as call rate, time history, and spectral distribution
- Produce histogram displays of event statistics

Event Analyzer applications include:

- Automated measurement and display of statistics such as call rate or peak frequency over large data sets
- Automated editing of extended sound data into individual sound files
- Efficient development of detection algorithms for different sound types in a data set

The Event Analyzer contains two modules – **event detection (EVDX)** and **event analysis (EVAX)**. It requires no additional hardware beyond the ability to record and digitize the input sound file. The program works as follows:

- The user creates a digital sound file of any length, using SIGNAL or one of the direct-to-disk recording programs from Engineering Design.
- The user configures the EVDX module to detect the desired sound type.
- The EVDX module processes the entire sound file automatically, detecting occurrences of the characterized event type.
- For each event, the EVDX module measures a suite of event parameters, including time of occurrence, event duration, principal event frequency, and FM frequency range), writes them to an event log file and optionally saves the detected event as a sound file.
- The EVDX module provides modeling and display tools to visualize and refine detection parameters.
- The EVAX module processes the event log file for statistical analysis and display, producing summary statistics and histogram displays for a variety of parameters over the entire data set.



## 2. Software Installation

---

### Installation Steps

---

Follow these steps to install the Event Analyzer.

1. Launch the file **evanal\_xxx\_setup.exe** (where **xxx** is the version number) by double-clicking on the filename in Windows Explorer. If you received a zip file, extract **evanal\_xxx\_setup.exe** from the zip file, then launch it.
2. Follow the installation prompts.
3. Install Event Analyzer in the same directory as SIGNAL. See "Directory Structure" below.

### Directory Structure

---

The Event Analyzer (EA) is a collection of SIGNAL command files, which are programs written in the SIGNAL programming language. Throughout this guide:

- **<sigwin>** represents the SIGNAL root directory, which is **c:\Program Files\Engineering Design\Signal 5** in Windows XP and **c:\Program Files (x86)\Engineering Design\Signal 5** in Windows Vista and later.
- **<sigdata>** represents the SIGNAL user data directory, which is **c:\Program Files\Engineering Design\Signal 5\userdata** in Windows XP and **c:\Engineering Design\Signal 5\userdata** in Windows Vista and later.

The EA installer creates three directories.

- Program files and read-only data files are installed in the **EA system directory** **<sigwin>\syscom\\_evanal**. This directory is managed by SIGNAL and **should not be altered by the user**.
- Program files shared with other SIGNAL event analysis programs are installed in the **Event analysis shared directory** **<sigwin>\syscom\\_ev\_shar**. This directory is managed by SIGNAL and **should not be altered by the user**.
- During EA operation, program output data such as event log files and output sound files are stored in the **EA user directory** **<sigdata>\evanal**. This directory might also contain input sound files for analysis. This directory belongs to the user, and its contents can be altered.

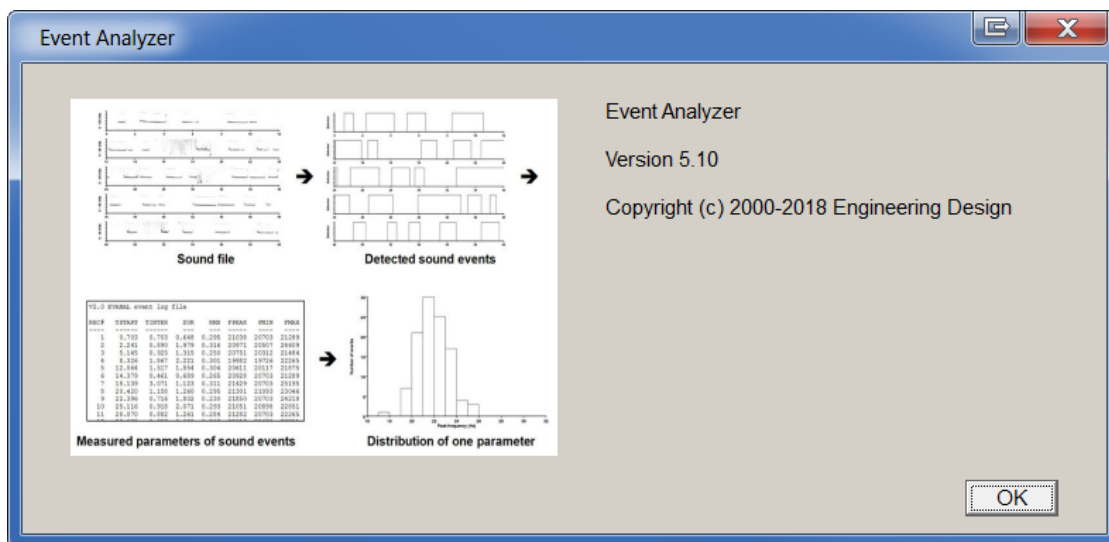
The following table summarizes the Event Analyzer directory structure, showing the directory location of various file types:

<b>File type</b>	<b>Suffix</b>	<b>Directory type</b>	<b>Directory</b>
Executable program	[none]	EA system Event detector shared	<sigwin>\syscom\_evanal <sigwin>\syscom\_ev_shar
Configuration	*.cfg	EA system	<sigwin>\syscom\_evanal
Event log	*.log	EA user	<sigdata>\evanal
Event sound files	*.sig	EA user	<sigdata>\evanal
Output image files	*.bmp	EA user	<sigdata>\evanal

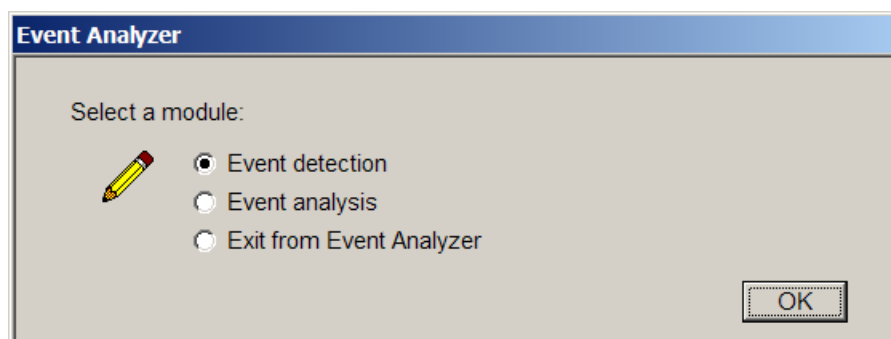
### 3. Launching the Event Analyzer

The Event Analyzer requires SIGNAL and the Event Detector module. To view your licensed modules, select Tools | License Info from the SIGNAL menu and make sure Event Detector is licensed.

To launch the Event Analyzer, select Signal|Event Analyzer from the SIGNAL menu or type "evanal" in the SIGNAL console window. The Event Analyzer will show the welcome screen:



followed by the main menu:



## 4. Event Detection (EVDX) Overview

---

The **event detection** module (also referred to in this guide as "EVDX") configures and operates the SIGNAL Event Detector. EVDX collects and manages detection parameters from the user, configures and runs the SIGNAL Event Detector, measures each detected event for quantities such as time of occurrence, event duration, interevent interval, and peak frequency, stores these measurements in a data file for statistical analysis, and optionally saves the detected events as SIGNAL sound files.

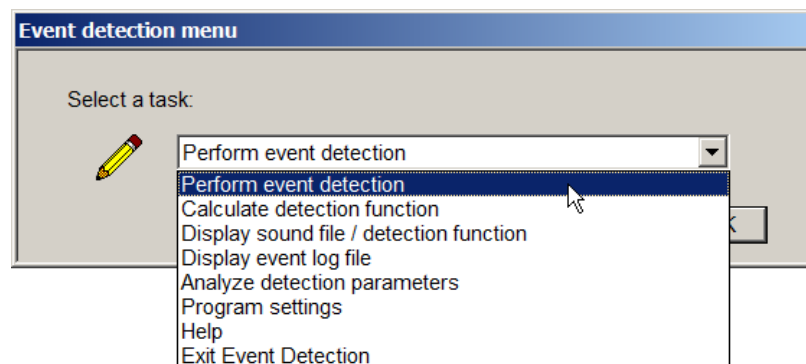
EVDX performs this process automatically on an entire sound file. Its input is a continuous digitized sound file and its outputs are the measurement data file and the optional event sound files. EVDX can replace a considerable amount of SIGNAL programming on the part of the user.

This section provides an overview of EVDX operation. Detection parameters, event detection, detection modeling, the detection function, and the event log file are discussed in later sections.

## 5. EVDX: Menu Overview

---

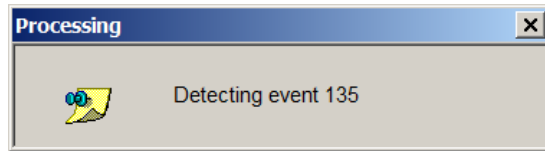
The EVDX menu contains all EVDX program operations. This section summarizes the menu selections. These are explained in detail in later sections, which are referenced in each description. Following is the EVDX main menu:



### Perform event detection

---

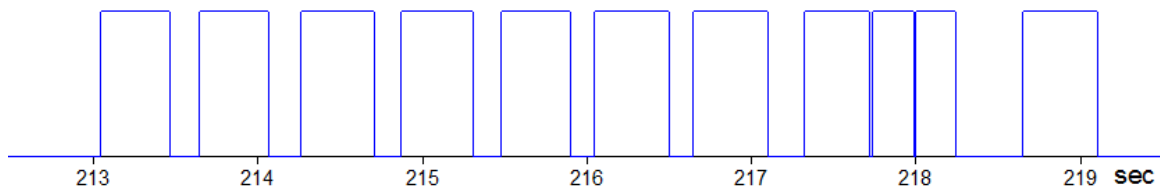
Detect vocalization events in the input file, measure event parameters (such as time of occurrence, event duration, and F0), write measurement data to the event log file, and optionally save detected events as individual sound files. The input sound file may be in SIGNAL, Wave, or AIFF file formats. See "EVDX: Event Detection" and "EVDX: Event Log File".



## Calculate detection function

---

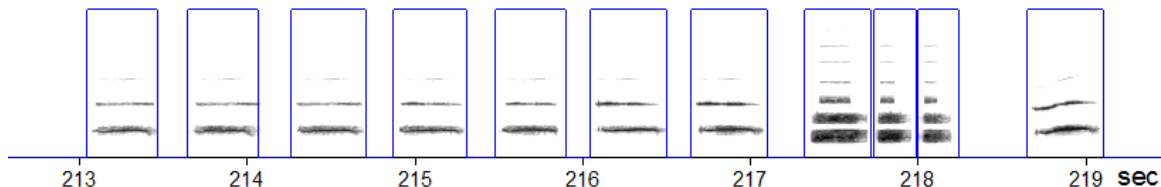
Calculate a **detection function** based on the event log file, which graphically displays the time location and duration of events. Using **Display sound file / detection function**, the user can display this function overlaid on the source waveform or spectrogram, to efficiently evaluate detector performance, diagnose problems, and manually edit individual events. See "EVDX: Detection Function".



## Display sound file / detection function

---

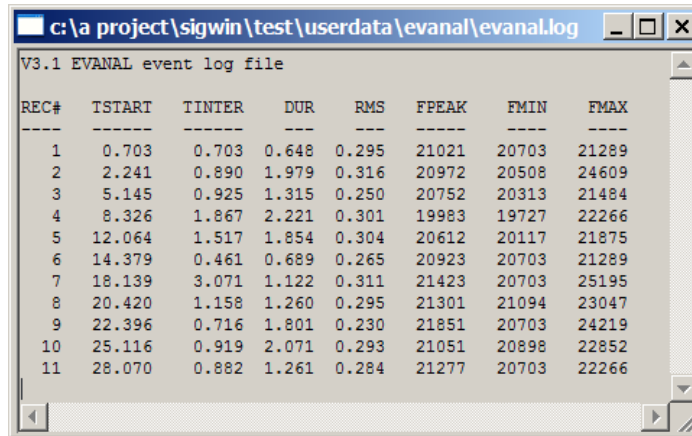
Produce a continuous display of the source sound file as a waveform or spectrogram, and optionally overlaid with the detection function. This is useful to check recording validity, verify the presence of desired acoustic events, and evaluate and troubleshoot detector performance. See "EVDX: Sound File / Detection Function".



## Display event log file

---

Display the event log file produced by **Perform event detection** in the EVDX menu. See "Event Log File".



V3.1 EVANAL event log file

REC#	TSTART	TINTER	DUR	RMS	FPEAK	FMIN	FMAX
1	0.703	0.703	0.648	0.295	21021	20703	21289
2	2.241	0.890	1.979	0.316	20972	20508	24609
3	5.145	0.925	1.315	0.250	20752	20313	21484
4	8.326	1.867	2.221	0.301	19983	19727	22266
5	12.064	1.517	1.854	0.304	20612	20117	21875
6	14.379	0.461	0.689	0.265	20923	20703	21289
7	18.139	3.071	1.122	0.311	21423	20703	25195
8	20.420	1.158	1.260	0.295	21301	21094	23047
9	22.396	0.716	1.801	0.230	21851	20703	24219
10	25.116	0.919	2.071	0.293	21051	20898	22852
11	28.070	0.882	1.261	0.284	21277	20703	22266

This selection displays the event log file produced by the **Perform event detection** menu choice.

---

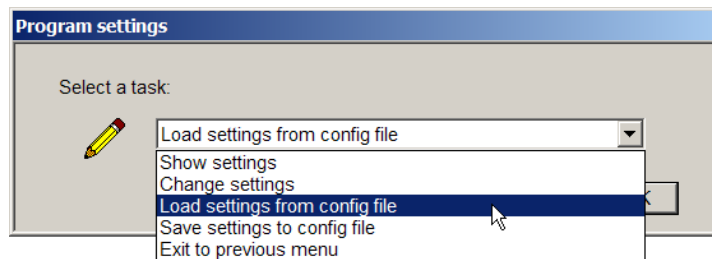
## Analyze detection parameters

Visualize the detection process by displaying the underlying functions for a specified segment of the source file. Refine the detection process by altering detection parameters and seeing the immediate impact on detection results. This is extremely useful in troubleshooting and refining detector performance.

---

## Program settings

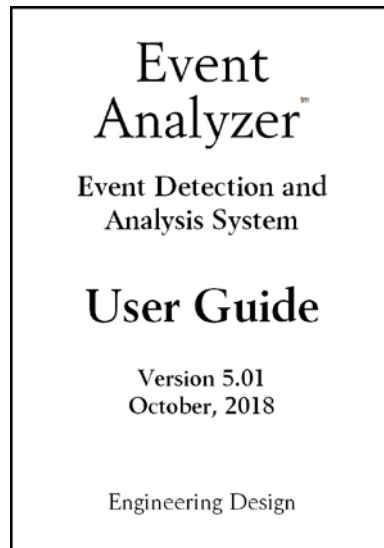
Event detection is configured by a number of parameters, referred to as **program settings**. This selection opens the Program Settings menu, which allows the user to view, change, load or save program settings. See "Program Settings".



---

## Help

Display the Event Analyzer User Guide (this document).



## Exit Event Detection

---

Exit the Event Detection module, back to the main Event Analyzer menu.

## 6. EVDX: Program Settings

---

Event detection is configured by a number of parameters, referred to as **program settings**.

### Configuration Files

---

Program settings are stored in disk files called **configuration files**. See "Show Settings" below for a sample configuration file.

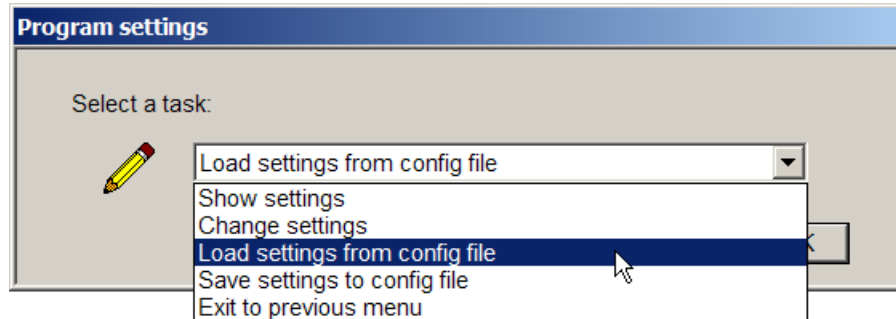
Program settings are automatically retained between Event Analyzer sessions in the working configuration files evd.cfg (event detection module) and eva.cfg (event analysis module). The user can also save settings to any number of custom configuration files and read them back on demand. These files must have the ".cfg" suffix so the program can find them. This allows the user to create different configuration files for different analyses, each containing the program settings for that analysis.

Configuration files have a prescribed format and **should not be edited directly by the user**. To alter or create configuration files, use operations from the Program Settings menu.

If either working configuration file is missing or unreadable, the Event Analyzer will instead load the corresponding **default configuration file**, evd\_default.cfg or eva\_default.cfg. The user can load these default files (see "Load Settings from File" below) to restore default program settings.

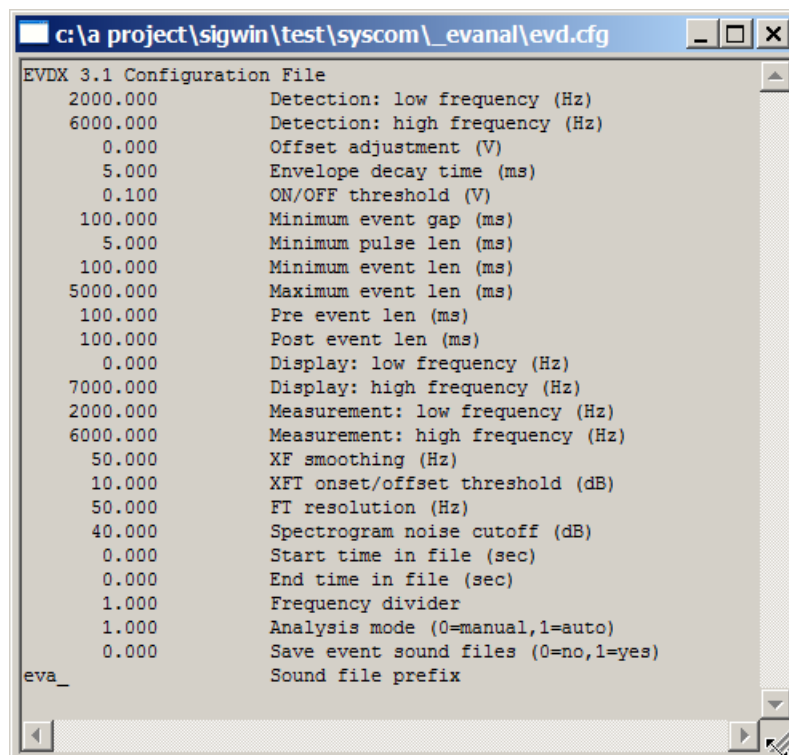
## Program Settings Menu

The Program Settings menu allows the user to display, change, load or save current program settings.



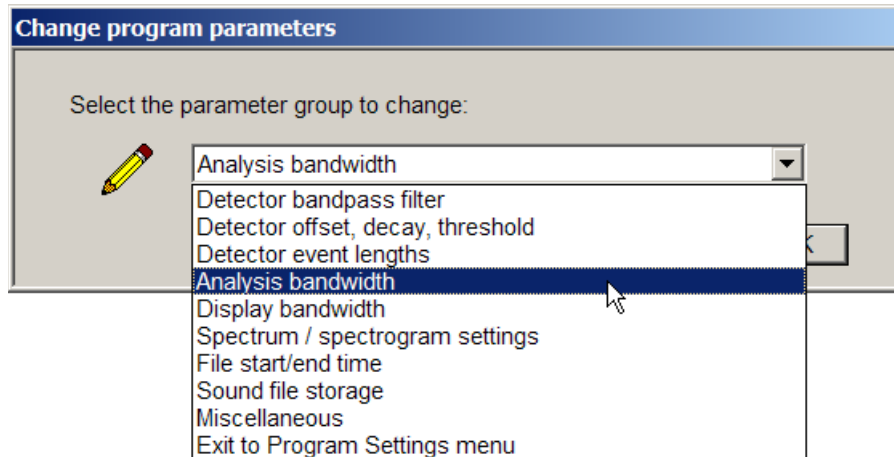
## Show Settings

This selection displays the current program settings, stored in the working event detection configuration file evd.cfg.

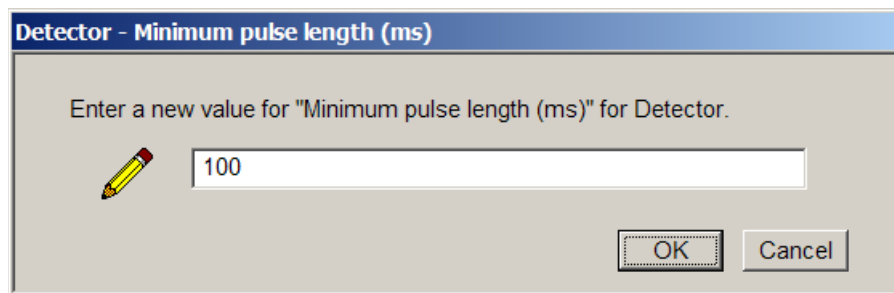


## Change Settings

This selection allows the user to alter individual parameter values. It displays a menu of parameter groups:

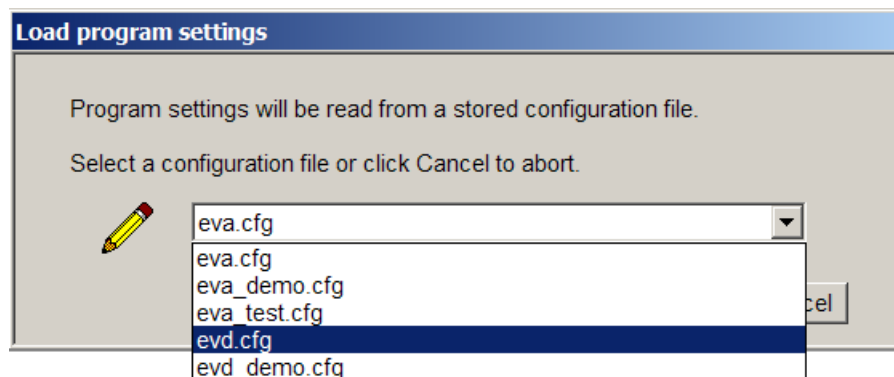


The user selects a group, then the program displays the parameters in that group, one by one, with their current value. The user can enter a new value or hit <enter> to leave the parameter unchanged. In the following example, the user changes minimum pulse length to 100 msec:



## Load Settings from File

This selection loads current program settings from a file specified by the user. Note that EVDX and EVAX maintain separate configuration files, and both file types will appear in the file list (see the figure). **Be sure to load the correct configuration file type** – or unpredictable behavior may result.

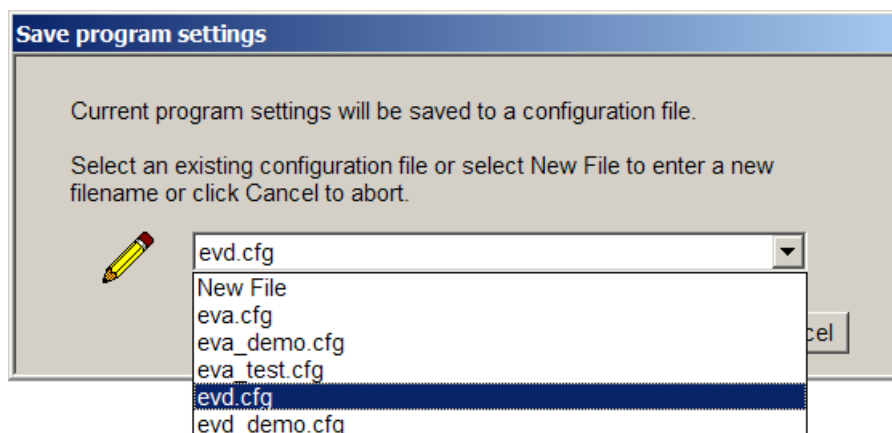




## Save Settings to File

---

This selection saves current program settings to a specified configuration file. Select an existing file or select "New File" and enter a filename when prompted. The filename must end with a ".cfg" suffix so the program will recognize it.



## Description of Parameters

---

Following is a description of EVDX configuration parameters. These parameters control event detection, display, and measurement, and other operating characteristics. **See the SIGNAL Event Detector User Guide for detailed discussion of detector parameters.**

**Note** that if the source file is frequency-divided (see "Adjusting Parameters for Frequency-Divided Sound Data" below), **all frequency parameters** (such as bandpass filter, analysis bandwidth, spectrum smoothing, etc.) should be divided by the frequency divider factor.

### Detector Bandpass Filter

**Low frequency cutoff** and **High frequency cutoff** set the lower and upper cutoffs of the bandpass filter in the event detector. These may be limited to some frequency range unique to the target signals, and can be unrelated to the analysis bandwidth over which the signal is measured (see Analysis bandwidth below).

### Detector Offset, Decay, Threshold

**Offset adjustment** is subtracted from the input signal in order to cancel DC offset.

**Envelope decay time** sets the decay (smoothing) constant used to calculate the signal amplitude envelope.

**ON/OFF threshold** sets the threshold amplitude level for onset and offset detection. Local (not necessarily event) onset occurs when the signal rises above this threshold and local offset occurs when the signal falls below this level. Threshold is normally set slightly above the ambient noise level in the input signal. This is a critical detection parameter. See the SIGNAL Event Detector User Guide.

## **Detector Event Lengths**

**Minimum pulse length** sets the minimum pulse length within an event. Shorter pulses are ignored as spurious noise bursts.

**Minimum event gap** sets the minimum separation between events. Longer separations are interpreted as the end of an event.

**Minimum event length** and **maximum event length** set the minimum and maximum event length, respectively, to reject unwanted events.

**Pre-event length** and **post-event length** specify the additional amount of signal to be appended before and after the detected event onset and offset, respectively.

## **Analysis Bandwidth**

**Low frequency cutoff** and **High frequency cutoff** set the frequency range over which power spectra and spectrograms are analyzed. These limits are often wider than detector bandpass filter limits, discussed above.

## **Display Bandwidth**

**Low frequency cutoff** and **High frequency cutoff** set the frequency range for displaying power spectra and spectrograms.

## **Spectrum / Spectrogram**

EVDX measures the peak frequency of each event, which is the frequency coordinate of the maximum amplitude value in the event's smoothed power spectrum. **Spectrum smoothing** specifies the spectral smoothing width in Hz.

**Spectrogram onset/offset threshold** specifies the threshold level used to calculate the minimum and maximum frequency (i.e., the bandwidth) of the detected event. Minimum frequency is the lowest frequency containing energy above the specified threshold. Threshold is specified in dB below peak spectrogram level, so reducing its value raises the threshold.

**Spectrogram resolution** specifies the frequency resolution of displayed spectrograms.

**Spectrogram noise cutoff** sets the lower intensity cutoff for spectrogram display. Components below this cutoff, such as noise, are not displayed. This is specified in dB below peak spectrogram level, so reducing its value will raise the cutoff and reduce the displayed noise in the spectrogram.

## **Operation Note**

When EVDX measures the minimum and maximum frequency of each detected event, three parameters help exclude non-event energy from this measurement. **Low frequency cutoff** and **high frequency cutoff** under "Analysis bandwidth" restrict the measurement range, which can be used to exclude high-intensity out-of-band non-event energy. And the measured min and max frequencies must exceed the amplitude threshold specified by **Spectrogram onset/offset threshold**, so as to exclude low-level in-band non-event energy.

### **File Start/End Time**

Detection and measurement can be restricted to a portion of the input sound file, for example, to analyze a portion of a long sound file. These limits are specified by **start time** and **end time** in seconds. To analyze the entire sound file, enter 0 for start and end time.

### **Sound File Storage**

EVDX can save each detected event as a SIGNAL sound file, to allow the user to examine the detected events or perform further measurement and analysis. This option is set by the **Save event sound files** parameter. Enter 1 to save sound files and 0 to decline the option.

Saved sound files are named *<prefix>nnnn.sig*, where *<prefix>* is a user-specified prefix (any number of characters) and *nnnn* is the 4-digit sequential event number, for example, *eva\_0001.sig*. The prefix is set by the **Sound file prefix** parameter and defaults to "eva\_".

See "EVDX: Event Detection" for details.

### **Miscellaneous**

Many bat detectors include a frequency divider circuit which alters the frequency but not the time scaling of the input signal. See "Adjusting Parameters for Frequency-Divided Sound Data" for details. The **Frequency divider** parameter allows the user to conveniently analyze sound recordings made through a frequency divider. It directs EVDX to adjust the frequency values in the event log file (peak, minimum, and maximum) to represent the true frequency of the original sound. For example, with this parameter set to 16, a frequency of 2000 Hz in the detected event will be stored in the log file as 32000 Hz.

The event detector normally operates in **automatic** mode, in which events are detected and measured without any significant screen display and with no user intervention. Alternately, the detector can operate in **manual** mode, in which intermediate detection and measurement steps are displayed on the screen for user examination. This is useful for monitoring and refining detector operation. This is set by the **Analysis mode** parameter. See "EVDX: Event Detection" below for details.

### **Exit to Program Settings menu**

Save all settings in the working configuration file, then exit to the Program Settings menu.

---

## **Adjusting Parameters for Frequency-Divided Sound Data**

Many bat detectors include a frequency divider circuit which allows ultrasonic signals to be heard and recorded on audio bandwidth equipment. Temporal scale is retained but the principal frequency component is detected (often via zero-crossing analysis) and resynthesized at some integral divisor of the original frequency. For example, with a frequency divider of 16, a 160 kHz tone is delivered by the bat detector as 10 kHz.

EVDX can process these signals, but some program parameters must be altered accordingly. Since the frequency spectrum of the input sound data is physically divided, **all EVDX frequency parameters must be divided accordingly**. For example, to impose a bandpass range of 18 – 26 kHz on an input signal that has been frequency-divided by 16, the bandpass

range in EVDX should be set to 1125 - 1625 Hz. The following parameters must be divided by the frequency divider factor:

- Detection: low frequency
- Detection: high frequency
- Display: low frequency
- Display: high frequency
- Measurement: low frequency
- Measurement: high frequency
- XF smoothing
- FT resolution

In addition, be sure to set the "Frequency divider" parameter to the frequency divider value. This will cause EVDX to rescale the frequency results stored in the event log file to reflect the original (undivided) physical frequency values. Finally, note that all spectrum and spectrogram displays produced by EVDX will be presented in frequency-divided units.

## 7. EVDX: Event Detection

---

The event detection module process an entire sound file automatically and delivers a measurement data file and optional event sound files. EVDX performs event detection by configuring and executing the SIGNAL Event Detector. For technical background on event detection, see the **SIGNAL Event Detector User Guide**. This section discusses EVDX operation.

### Detection Parameters

---

Event detection parses the recorded data stream into individual vocalization events, according to user-specified spectral, temporal and amplitude parameters, such as spectral bandwidth, temporal duration, and amplitude threshold. **Detection parameters** are selected so as to define the acoustic boundaries of a target event while excluding non-target sounds such as animal movement and external noise. For example, detection bandwidth is selected so as to include significant spectral energy bands of the event, but exclude low-frequency environmental noise, and amplitude threshold is typically set to just above the ambient noise level in the signal. Following is a list of all detection parameters used by EVDX.

- Low frequency detection limit
- High frequency detection limit
- Amplitude offset adjustment
- Amplitude envelope decay time
- Onset/offset amplitude threshold
- Minimum event gap
- Minimum pulse length
- Minimum event length
- Maximum event length

Pre event length  
Post event length

EVDX detection parameters are stored in **configuration files**, along with other EVDX program settings, and can be modified at any time. See "Program Settings".

Although detection parameters are maintained by EVDX, the SIGNAL Event Detector uses them to perform the actual detection. See the **SIGNAL Event Detector Guide** for discussion of these parameters and general detection strategies.

---

## Program Queries

EVDX queries the user for the names of the input sound file, the output log file, and optionally the directory to receive stored event sound files.

The **input sound file** may be in the SIGNAL, Wave or AIFF file format, and is typically produced by a direct-to-disk recording program, such as CBDisk, DartDisk, DTDisk, NIDisk or WaveDisk, or other commercial program. File size is virtually unlimited.

EVDX stores measured event parameters in a text data file called the **event log file**, one data record per event (see "EVDX: Event Log File"). EVDX will warn if this file already exists before overwriting it.

EVDX can optionally store every detected event as an **event sound file**. See "Event Sound Files" below.

---

## Event Sound Files

Detected events can optionally be saved to individual sound files. This can be useful to verify detection results or perform more detailed acoustical analysis using SIGNAL.

To enable this option, set **Save event sound files** to 1 in the Sound File Storage section of Program Settings. EVDX will query for the directory to receive these files.

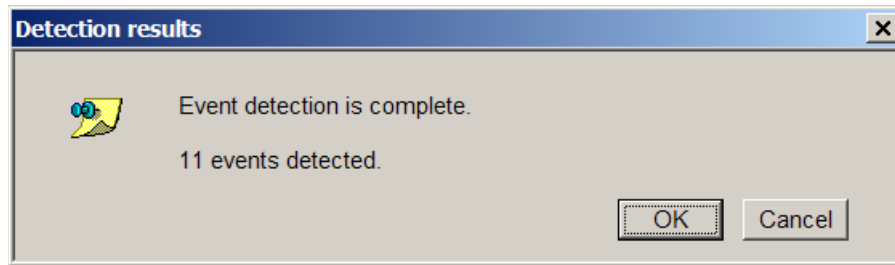
Event sound files are stored in the SIGNAL file format. They are named *<prefix>nnnn.sig*, where *<prefix>* is a user-specified prefix (any number of characters) and *nnnn* is the 4-digit sequential event number, beginning with 1 for each detection run. The prefix is set by the **Sound file prefix** parameter, also in the Sound File Storage section of Program Settings. Thus with the default prefix "eva\_", successive files will be named *eva\_0001.sig*, *eva\_0002.sig*, etc. To avoid accidental overwrites, EVDX warns the user before overwriting any of these files. The overwrite can be approved or the write cancelled.

By default, sound files are stored in the EA user directory (see "Software Installation"). However, it is recommended that the sound files from each detection run be stored in a different, uniquely-named directory, such as *c:\eva\_data\trial\_056*. The event log file might be stored in the same directory for security.

## Detection Report

---

At the end of detection, EVDX reports the total number of events detected:



## 8. EVDX: Manual Detection

---

Event detection normally operates in **automatic** mode, detecting and measuring events in the input sound file without user intervention. EVDX also provides a **manual** mode for monitoring, diagnosing, and refining detector performance. In manual mode, intermediate detection and measurement results are displayed and the user can confirm or adjust them.

Manual mode can also be used to analyze difficult source material in a semi-automated manner. An example would be noisy source material on which the detector cannot perform consistent detections automatically, so that each detection must be checked and possibly adjusted. This mode is slower but, because it still provides the measurement tools and data structures of EVDX, it is still much more efficient than a purely manual analysis.

To operate the detector in manual mode, open the Miscellaneous menu in the Program Settings menu, and set the **Analysis mode** parameter to Manual. Manual mode provides four screens. These are described on the following pages.

## Onset and Offset Times

The first screen shows the detected event and allows the user to alter event onset and offset times, to correct for detector errors. See Figure 1. The user can accept the time boundaries of the event as detected, modify them by entering new start and/or end times, or discard the event entirely (in case of a spurious detection). The event is then redisplayed with new boundaries and the cycle can repeat.

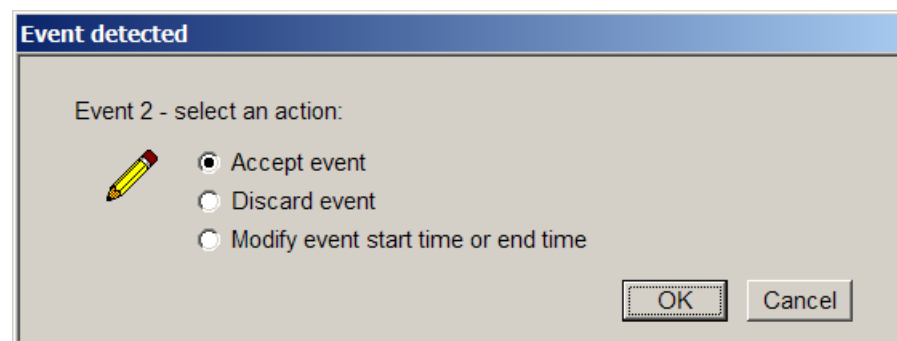
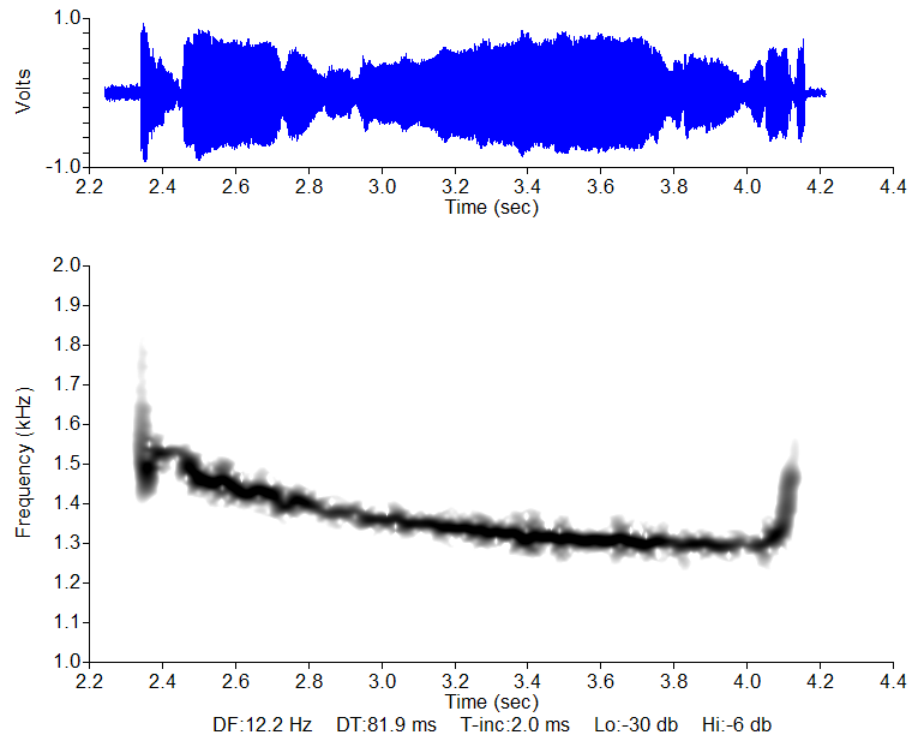


Fig 1. Manual detection: confirm or adjust event start and end times

## Peak Frequency

---

The second display shows the power spectrum of the event with the measured peak frequency. See Figure 2. The user can accept or change the peak value. To modify the value, move the cursor to the desired peak frequency value on the horizontal axis and hit <esc>.

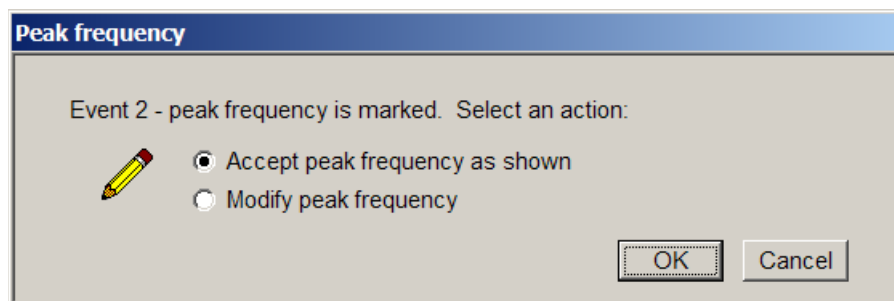
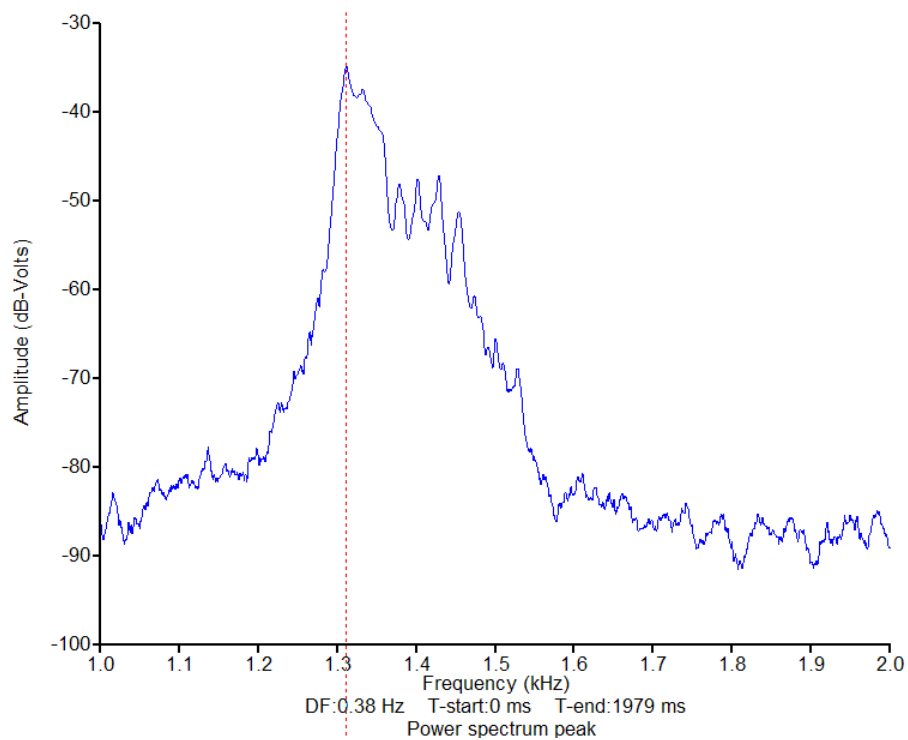


Fig 2. Manual detection: confirm or adjust peak frequency



## Minimum and Maximum Frequency

The third display shows the detected event with measured minimum and maximum frequency indicated over the event spectrogram. See Figure 3. The user can accept these values or modify them by moving the horizontal cursor to the desired minimum or maximum frequency value on the vertical axis and hitting <esc>.

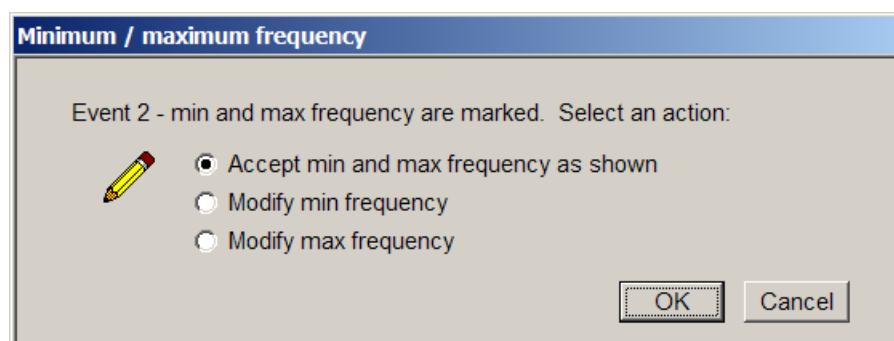
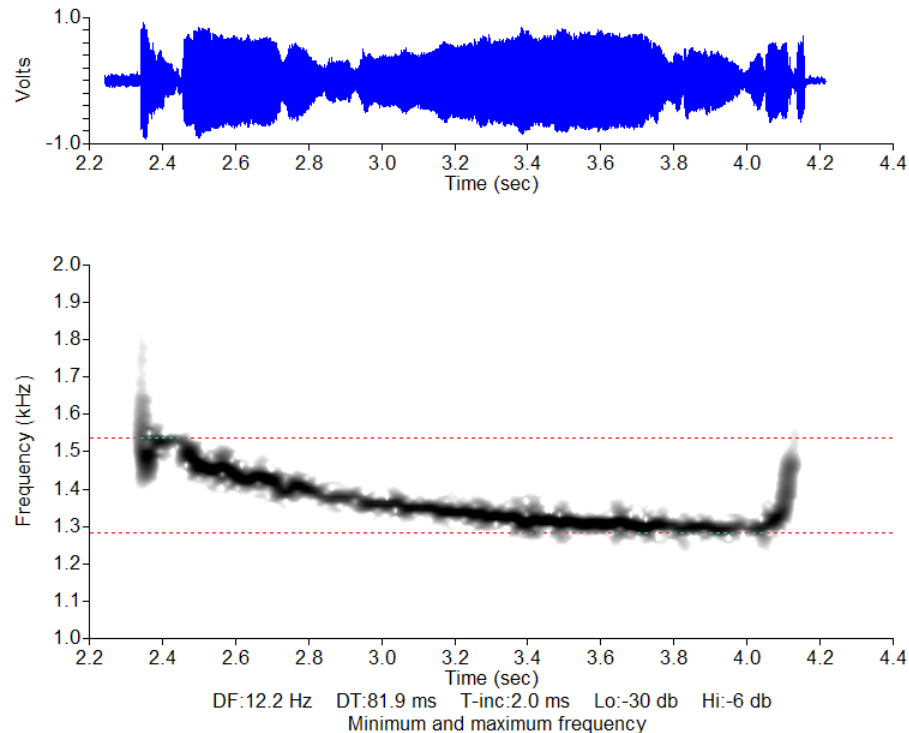


Fig 3. Manual detection: confirm or adjust min and max frequency

## Event Parameter Summary

---

The final screen summarizes the measured event parameters, including any changes made in the previous screens. See Figure 4.

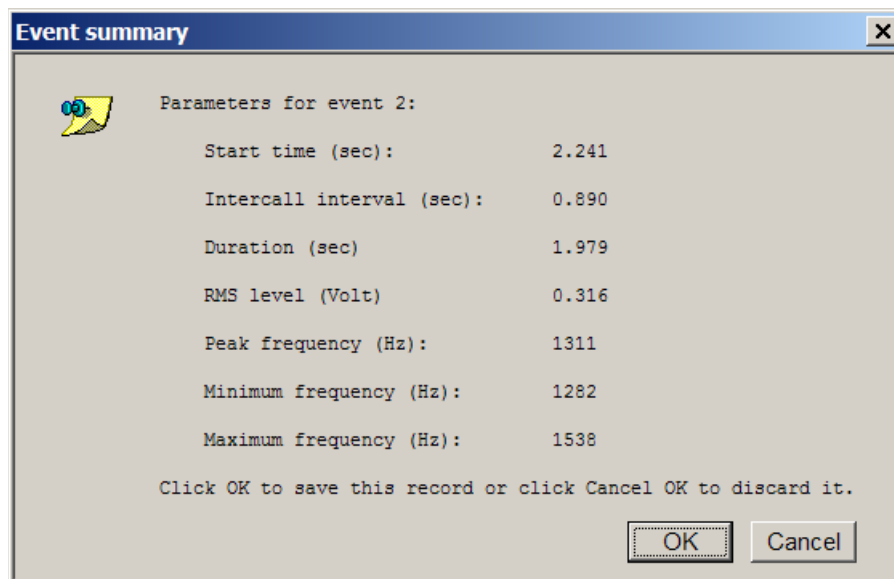


Fig 4. Manual detection: parameter summary

## 9. EVDX: Detection Model

---

EVDX provides an interactive tool called the **detection model (DM)** for refining the detection process on-screen. The detection model displays the component detection functions for a specified segment of the input sound file. The user can alter detection parameters, recalculate the detection functions, and observe the effect. This can be very useful in diagnosing and refining detector performance.

To access the DM, select **Analyze detection parameters** from the EVDX menu, select a source file to analyze, and specify the analysis segment within the file. Analysis segments are normally quite short, so that fine time and amplitude details can be observed.

### Detection Mathematics

---

Following is an outline of the basic steps and parameters by which SIGNAL and the EVDX perform event detection. See the **SIGNAL Event Detector User Guide** for a complete discussion of SIGNAL detection mathematics, including all steps, constraints and parameters.

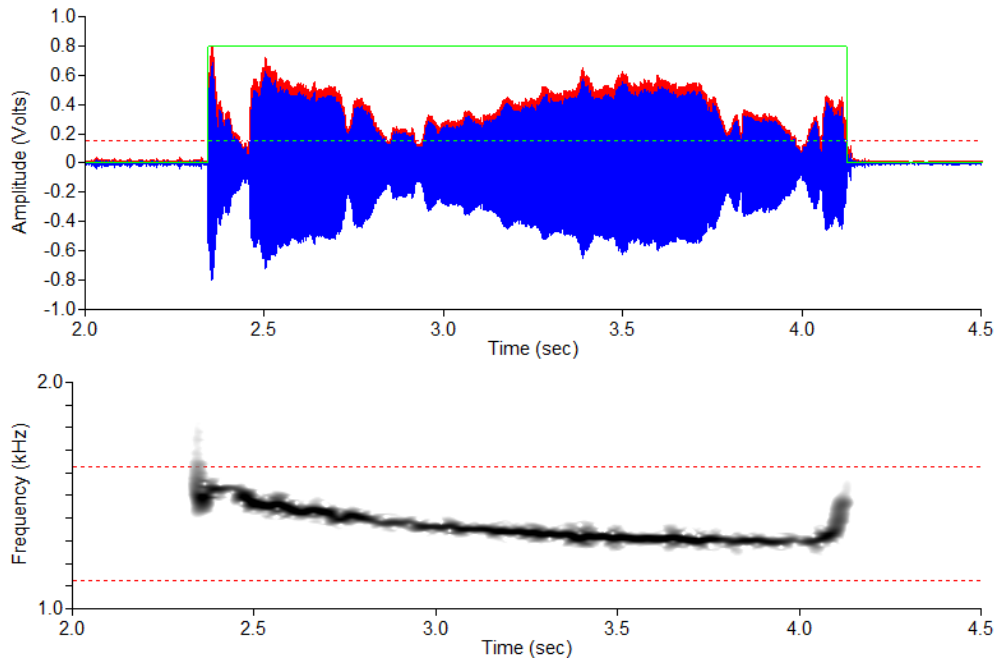
- The input signal is bandpass filtered according to the **lower bandpass** and **upper bandpass** frequency limits.
- The filtered waveform is converted to an amplitude envelope according to the **envelope decay** time.
- The envelope amplitude is compared to the onset/offset **threshold** level, producing a "logic" signal that is ON when the amplitude exceeds the threshold and OFF otherwise.
- ON and OFF periods are compared to the detector length constraints. ON periods shorter than **minimum pulse length** are treated as OFF periods (to reject spurious noise spikes) and OFF periods longer than **minimum event gap** serve as event separators, i.e., they terminate the current event. Several other length constraints are also applied.

---

## Detection Model Display

---

The detection model display is a split-screen. See the figure. The lower half shows the signal spectrogram with bandpass filter limits marked by red dotted lines. The upper half shows the signal waveform, **filtered per the current bandpass limits**, overlaid with the derived functions representing the detection steps (see "Detection Mathematics" above). The graph shows the **bandpass-filtered event waveform in blue**, its **amplitude envelope in red**, the **detection threshold in dotted red**, and the **detection function in green**. In the figure, the threshold is close to 0 and nearly overlays the X-axis. The green detection function represents detector output, with its rising edge at the detected event onset (TSTART) and the falling edge (TSTART + DUR) at the detected event offset. If no event were detected, the detection function would be zero everywhere.



This display can be used to evaluate detector success, i.e., whether the detection function properly encompasses the event (as in the figure), or only part of the event, or extends unnecessarily beyond the actual event. The detection model allows the user to adjust detection parameters – such as envelope decay, detection threshold, or minimum event gap – to correct these conditions. The model recalculates all detection functions and redraws the display after each parameter change. The user can save the modified parameters if satisfied, or discard them and exit or try again.

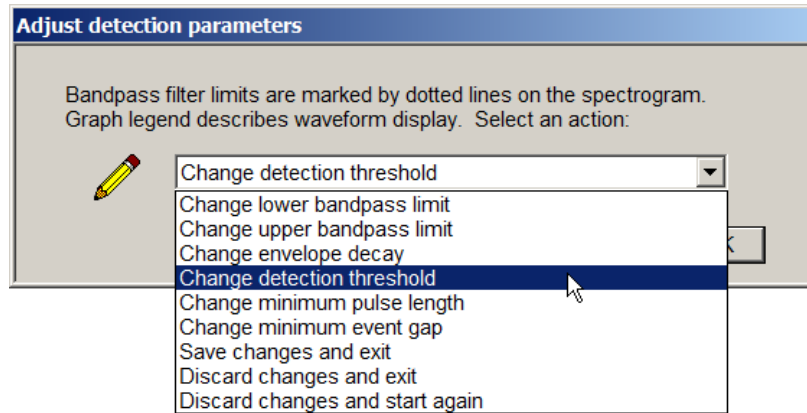
## Detection Model Menu

---

The detection model allows the user to alter the following parameters and see the immediate effects on the detection process:

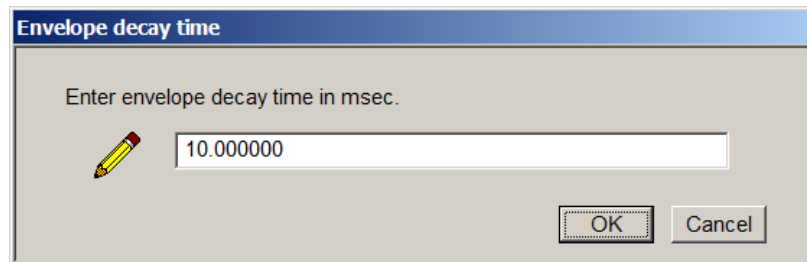
- Low frequency detection limit
- High frequency detection limit
- Amplitude envelope decay time
- Onset/offset amplitude threshold
- Minimum pulse length
- Minimum event gap

These parameters are accessed from the DM menu:



**Lower bandpass limit** and **upper bandpass limit** are altered using a crosshair cursor on the spectrogram display. Current bandpass filter limits are marked with dotted red lines. Bandpass limits should 1) exclude noise and non-target frequency components and 2) include some frequency band that is fairly unique to the target signal and sufficient in amplitude for the detector to recognize.

**Envelope decay** is adjusted by entering a new value numerically as shown. Excessive decay causes envelope "draping" after the envelope and artificially extends events or causes them to fuse with adjoining events. Insufficient decay causes events to break apart, i.e., not fuse with themselves.



**Detection threshold** is adjusted using a crosshair cursor on the waveform display. Click in that display to locate the cursor there. Threshold should be set just above the visible noise threshold. Too low a threshold causes adjacent events to fuse, and too high a value causes individual events to break apart.

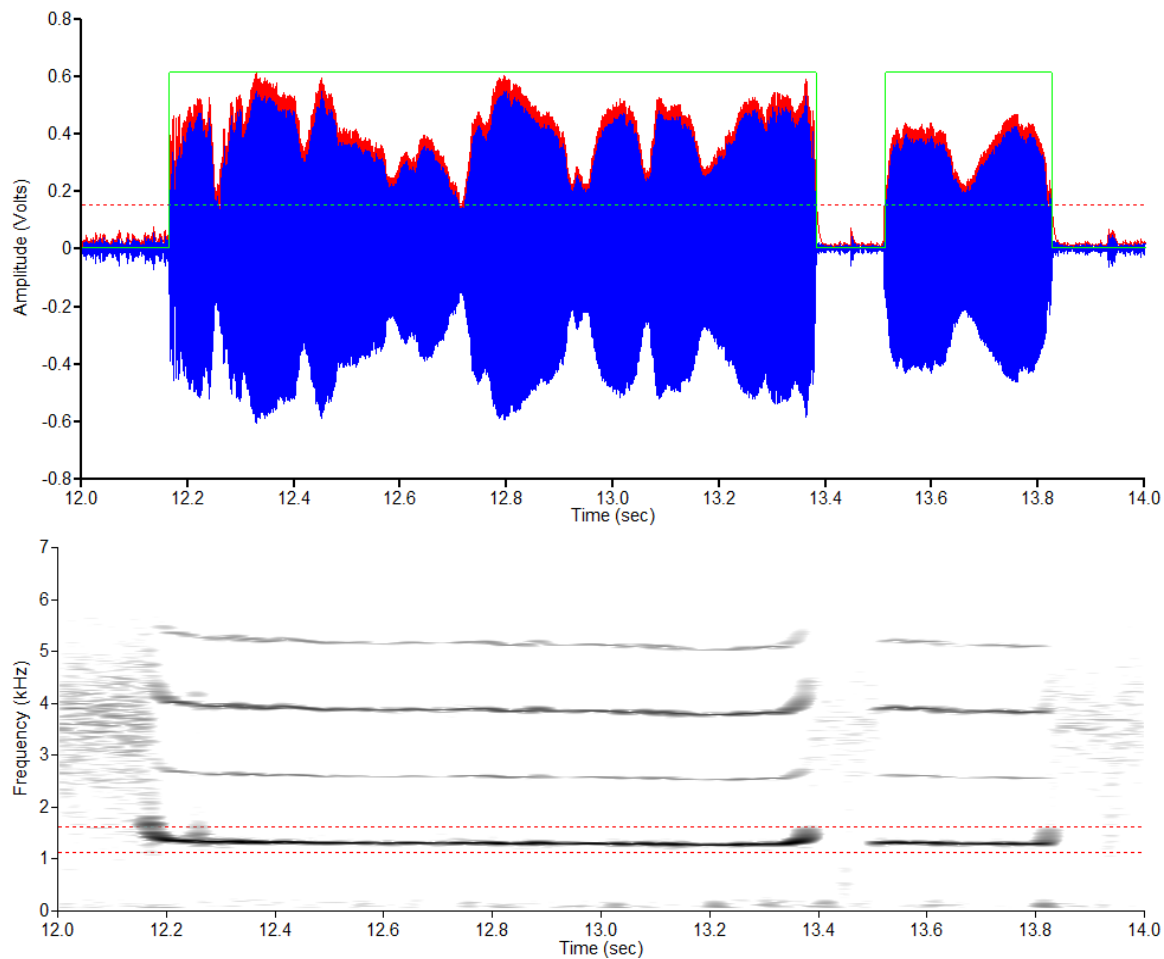
**Minimum pulse length** and **minimum event gap** are adjusted by entering new values numerically, similar to envelope decay. Too large an event gap causes adjacent events to fuse and too small a value causes individual events to break apart.

After all desired changes have been made, select **Save changes and exit** to keep the new detection parameters and return to the EVDX menu, or **Discard changes and exit** to restore initial parameter settings and exit, or **Discard changes and start again** to restore initial parameter settings and try the adjustment process again.

## Detection Model Example

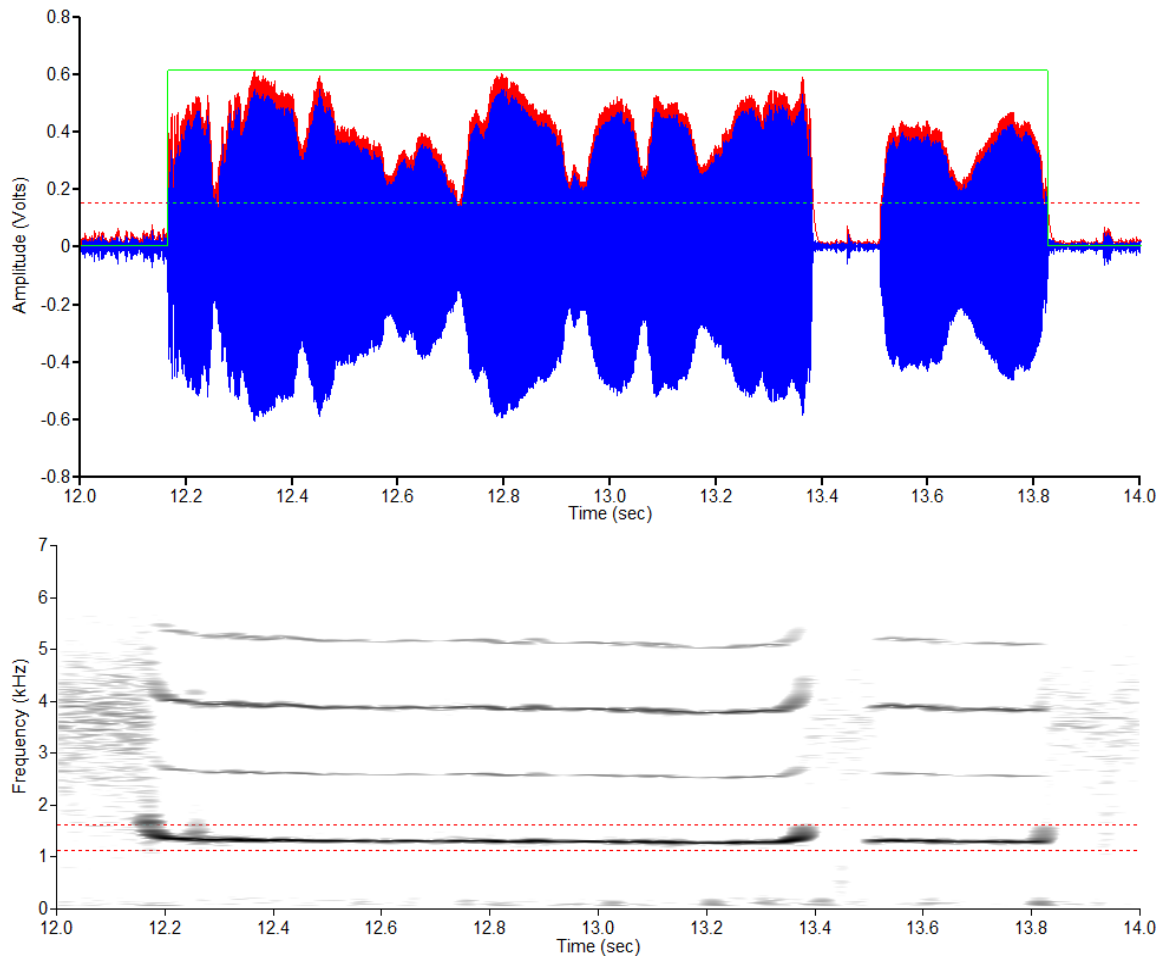
---

The following figure shows a detection model analysis of 2 seconds of `evanal_demo.sig` (12 – 14 seconds). For this example, the detection parameters have been set non-optimally, and the detector has incorrectly detected the two pulses comprising one event as two separate events, as shown by the green detection function.



The two pulses in the event have been detected as separate events because **minimum event gap** (set to 100 msec) is too small, causing the intra-event space to be recognized as an event separator. See the **SIGNAL Event Detector User Guide** for technical background on detection parameters.

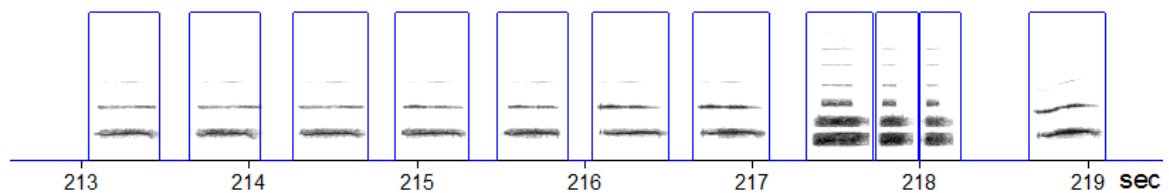
Increasing event gap to 400 msec produces the following display:



## 10. EVDX: Detection Function

### Detection Function

The **detection function** is a logic function that graphically represents the time location of every detected event. It is "high" for the detected duration of each event and "low" everywhere else. It is generated directly from the event start times and durations in the event log file. The following fragment shows the detection function (in blue) in relation to the underlying sound events. In this example, the detector has successfully detected each event.



The detection function graphically displays the detector's temporal performance, showing when events were detected and of what duration. It allows the user to quickly "read" the detector's behavior in order to evaluate overall performance and spot and diagnose problems. For example, the detection function might reveal any of the following behaviors: valid events are being missed (indicating that detection limits are too strict), spurious events are being detected (limits are too tolerant), sub-events are being separated (minimum event gap is too short), distinct events are being linked (minimum event gap is too long).

## Example

---

Figures 7 and 8 show a sound file and the corresponding detection function. These displays were produced by the STRIP command – see "Operating Hints" below. Here the detector has operated almost perfectly – the detection function includes every sound event and no spurious events. Comparing the two figures illustrates several aspects of detector behavior:

- The two sub-events at 12-14 secs have been linked into one detected event, while the event at 14-15 secs has been detected as a separate event. This is controlled by the "minimum event gap" parameter.
- Spurious energy at 16-18 secs has been rejected.
- Sub-events at 22-24 secs have been correctly linked into one detected event.
- Sub-events at 25-27 secs have been correctly linked into one detected event.

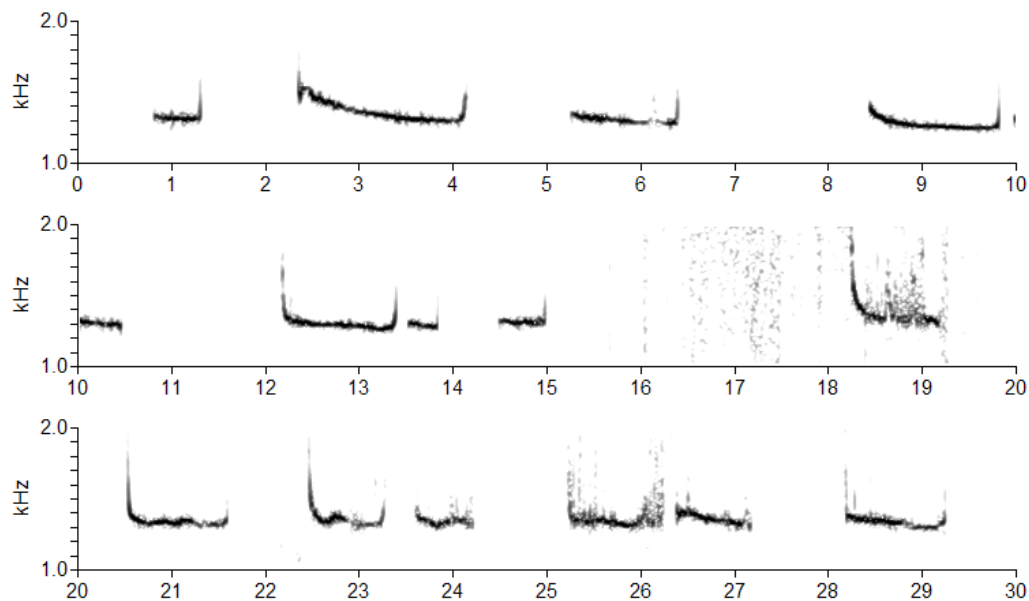


Fig 7. Sound file



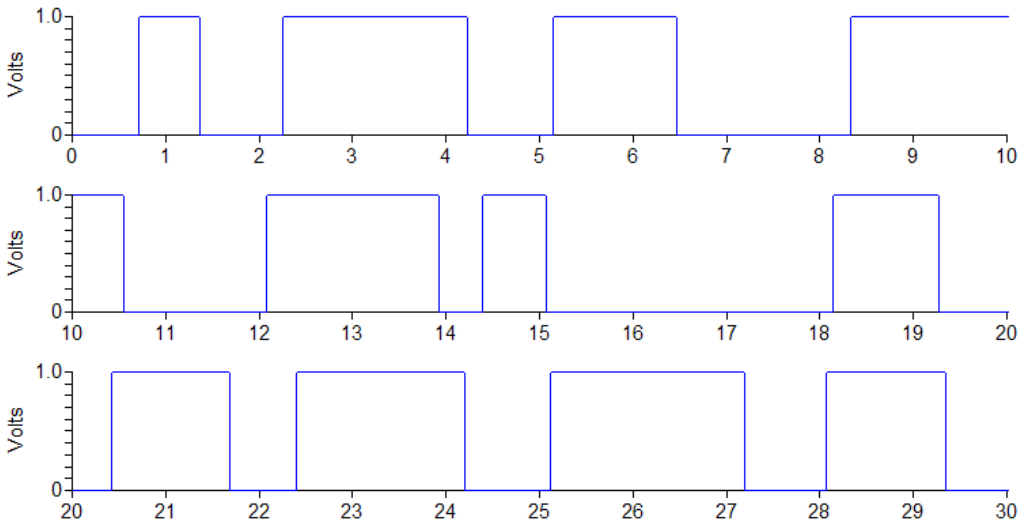


Fig 8. Detection function

## 11. EVDX: Sound File / Detection Function

---

### Sound File / Detection Function Display

---

The Event Analyzer can display a continuous waveform or spectrogram of a recorded sound file and/or detection function. This is useful to check recording validity, verify the presence of desired acoustic events, and evaluate and troubleshoot detector performance. To access this feature, select **Display sound file / detection function** from the EVDX menu.

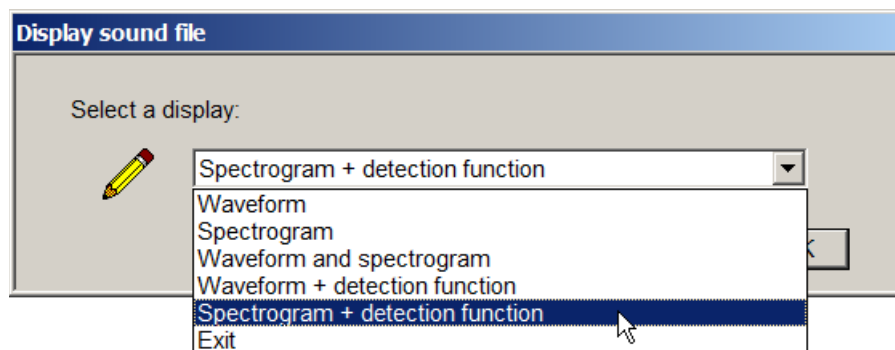
The display comprises multiple horizontal strips per screen and uses as many screens as necessary to cover the entire sound file. By default, there are 6 strips of 10 seconds each, for 60 seconds per screen, but the number and duration of strips can be varied via **No. spectrogram strips per page** and **Spectrogram strip duration**, respectively, in the **Spectrum / spectrogram** section of the Program Settings menu.

Each screen is automatically saved to an image file for later examination or printing. Files are saved to the Event Analyzer user directory (see "Software Installation").

### Display Types

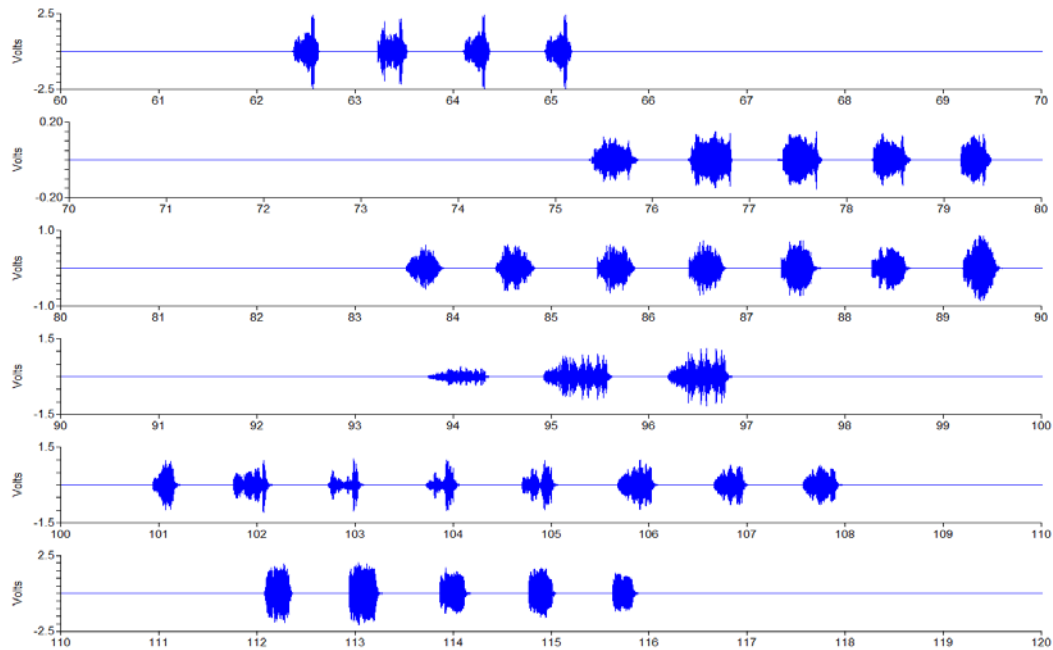
---

The **Display sound file / detection function** menu offers several displays:

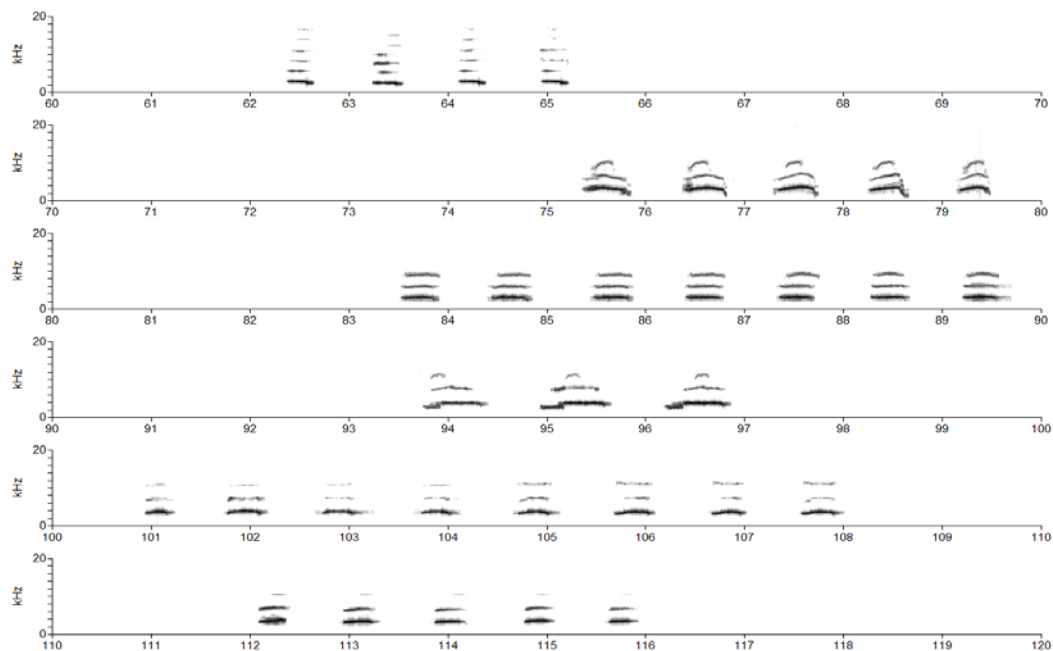


These menu choices are illustrated below. After selecting the display type, the user selects the sound file to display and optionally the file containing the detection function. All figures are based on the same source file segment. The continuous display is generated by the STRIP command in SIGNAL. See the **SIGNAL Reference Guide** for technical background on STRIP.

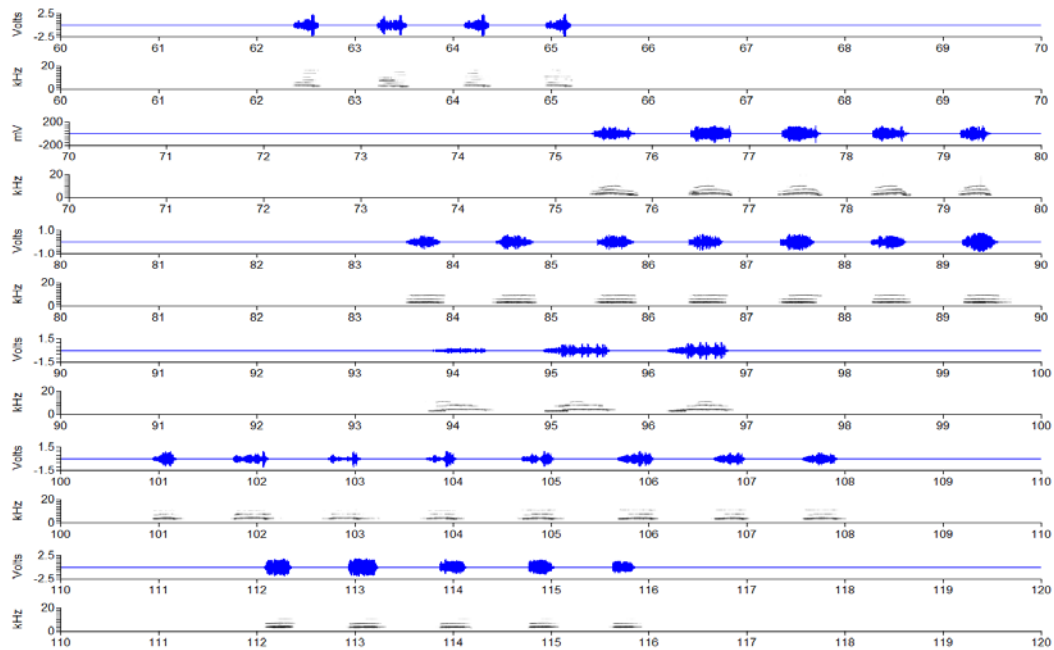
**Waveform** displays the signal's time waveform. Note that **vertical graph range may vary between strips** because each strip is calculated independently. This can also be used to display the detection function by itself (see also overlay displays below).



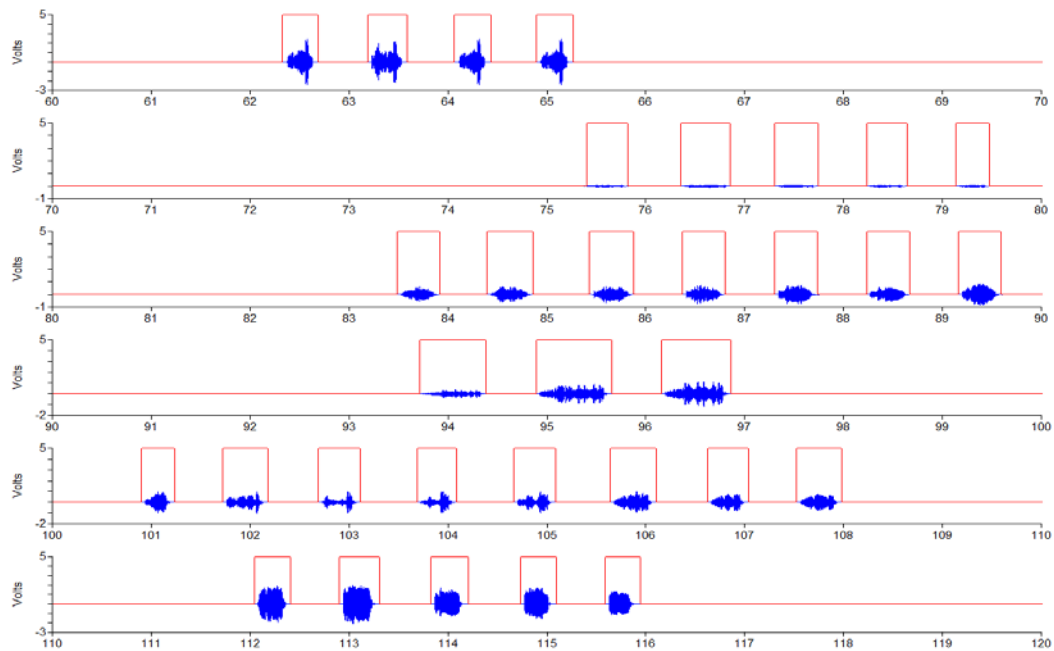
**Spectrogram** displays the signal spectrogram. Frequency range is determined by **Low frequency cutoff** and **High frequency cutoff** in the Display Bandwidth section of "Program Settings".



**Waveform and spectrogram** displays the signal waveform and spectrogram, time-aligned, strip by strip. Considerations noted for waveform and spectrogram displays above apply here.



**Waveform + detection function** displays the detection function overlaid on the signal waveform.file. This is useful for evaluating overall detector performance and identifying any incorrect detections. Detections can be corrected directly from this screen – see below.

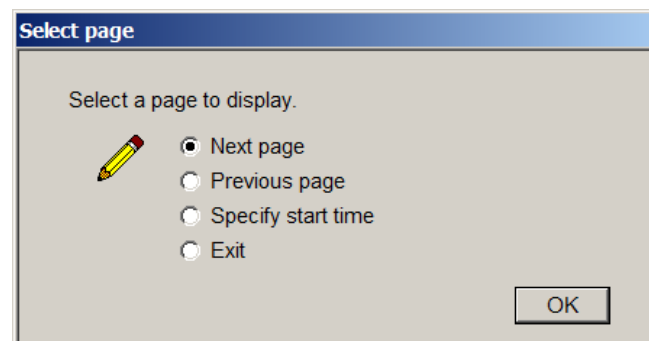


**Spectrogram + detection function** displays the detection function overlaid on the signal spectrogram. This display is also used to evaluate detector performance and identify incorrect detections.



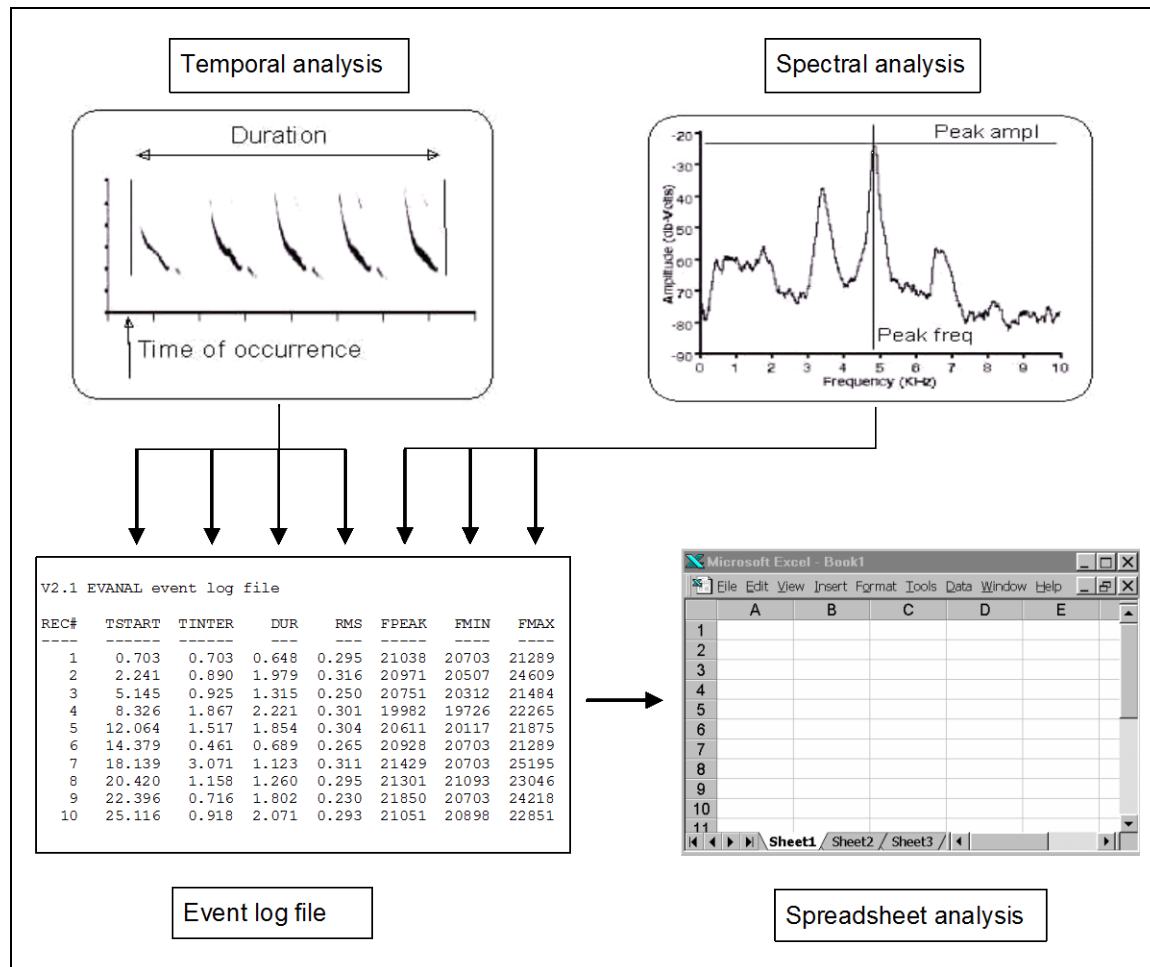
## Navigating the Screen Display

By default, the Event Analyzer proceeds screen by screen through the entire sound file. However, the user can also navigate forward and backward, or jump to a specified time location in the sound file.



## 12. EVDX: Event Log File

After measuring an event, EVDX stores the measured parameters – such as time of occurrence, event duration, and peak frequency – in a data file called the **event log file**. This file is normally processed by the **event analysis** module, but it can also be exported for analysis by spreadsheet and statistical programs. This flow is illustrated in the following diagram:



## Log File Structure

---

The event log file is an ASCII text file, and can be examined using a text editor or word processor. **The log file should not be altered by the user except as described here,** because the **event analysis** module requires the log file in its original format.

The log file begins with a **header section**. The first line identifies the file and the EVDX version that produced it. The last number on this line ("4" in the sample below) specifies the total number of lines in the header. The user can insert any number of documentary lines immediately after the first line as long as line count in the header line is adjusted appropriately.

The header is followed by the **data section**, which contains one line for each detected event. Each line contains a sequence of fields, and each field corresponds to one measurement. The following section describes these fields.

## Log File Fields

---

Each line in the log file is referred to as a **record** and corresponds to one event. The following table summarizes the fields in each record. The **field label** appears at the top of each column in the file.

<u>Field label</u>	<u>Contents</u>
REC#	Record (event) number
TSTART	Event onset time re sound file origin (sec)
DUR	Event duration (sec)
TINTER	Interval between this event and previous event (sec)
RMS	RMS amplitude level (Volts)
FPEAK	Peak frequency (Hz)
FMIN	Minimum frequency (Hz)
FMAX	Maximum frequency (Hz)

Following are comments on individual fields.

**Record number (REC #)** the sequential record (or event) number, beginning with 1.

**Event start time (TSTART)** the start time of the detected event in seconds, relative to the beginning of the input sound file.

**Event duration (DUR)** the duration of the detected event in seconds, between detected onset and offset, **including pre- and post-event extensions**, as specified by **Pre-event length** and **Post-event length** parameters (see "Program Settings").

**Inter-call interval (TINTER)** the time in seconds between the start of the current event and the end of the previous event.

**RMS level (RMS)** the RMS (root-mean-square) intensity level of the detected event in Volts. This value is calculated over the entire event duration, and therefore can vary with detection parameters such as pre-event and post-event extensions.

**Peak frequency (FPEAK)** the frequency value in Hz at which the event power spectrum achieves its maximum value. The power spectrum is smoothed per the **XF smoothing** parameter before calculating this value.

**Minimum frequency (FMIN), maximum frequency (FMAX)** the lowest and highest frequency values that exceed a specified intensity level. Measured from the event spectrogram. Limited to the bandwidth by **Measurement: low frequency** and **Measurement: high frequency**, to exclude out-of-band non-event energy. The **XFT onset/offset threshold** parameter specifies a minimum intensity level, re peak spectrogram level, to exclude low-level non-event energy.

## Sample Event Log File

---

Following is a sample event log file.

V3.1 EVANAL event log file							
REC#	TSTART	TINTER	DUR	RMS	FPEAK	FMIN	FMAX
----	-----	-----	---	---	-----	-----	-----
1	0.703	0.703	0.648	0.295	21021	20703	21289
2	2.241	0.890	1.979	0.316	20972	20508	24609
3	5.145	0.925	1.315	0.250	20752	20313	21484
4	8.326	1.867	2.221	0.301	19983	19727	22266
5	12.064	1.517	1.854	0.304	20612	20117	21875
6	14.379	0.461	0.689	0.265	20923	20703	21289
7	18.139	3.071	1.122	0.311	21423	20703	25195
8	20.420	1.158	1.260	0.295	21301	21094	23047
9	22.396	0.716	1.801	0.230	21851	20703	24219
10	25.116	0.919	2.071	0.293	21051	20898	22852
11	28.070	0.882	1.261	0.284	21277	20703	22266



## 13. EVDX: Tutorial

The following tutorial detects all events in the sample sound file **evanal\_demo.sig** and writes event data (such as event duration and time of occurrence) to the event log file **evanal.log**.

**Note 1:** Be sure to review "Directory Structure" in this guide before running this tutorial, to understand where Event Analyzer components are stored.

**Note 2:** If desired, you can examine output files while running through the tutorial:

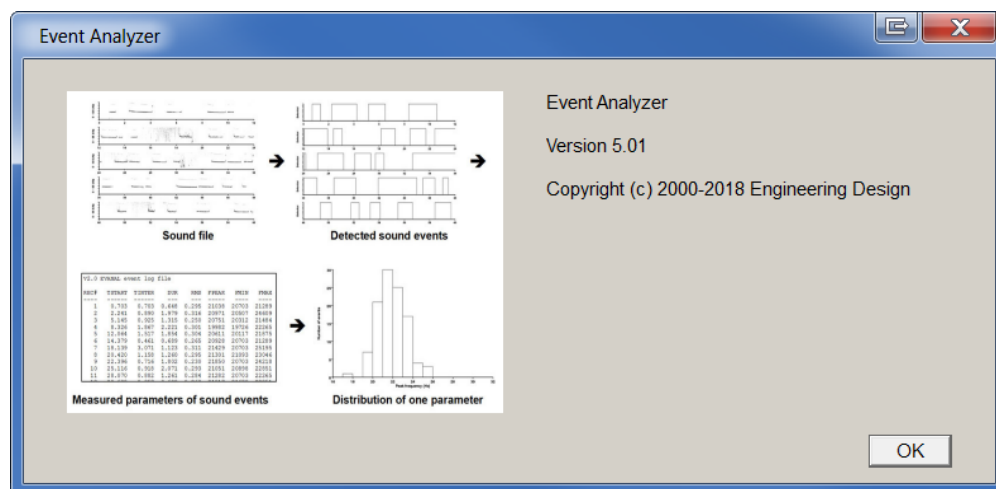
- Output data – including the log file, image files, and event sound files – are stored in the **EA user directory**, which is **c:\Engineering Design\Signal 5\userdata\evanal** in Windows Vista and later (see "Directory Structure").
- Event Detection reference files – such as demo signals and the reference demo log file – reside in the **EA system directory**, which is **c:\Program Files (x86)\Engineering Design\Signal 5\syscom\evanal** in Windows Vista and later (see "Directory Structure").

**Note 3:** If necessary, you can always restore EVDX default program settings as follows.

Select "Event detection" at the main Event Analyzer (EA) menu, select **Program settings**, then **Load settings from config file**, then "evd\_default.cfg" from the file list.

### Launch the Event Analyzer

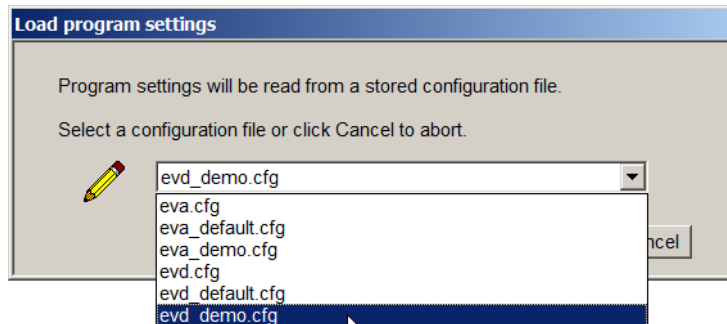
1. Start SIGNAL.
2. Select Signal|Event Analyzer from the menu or type "evanal" in the console window to launch the Event Analyzer.



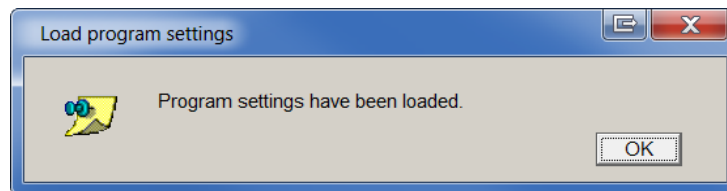
3. Select "Event detection" from the Event Analyzer menu.

## Set parameters for Event Detection

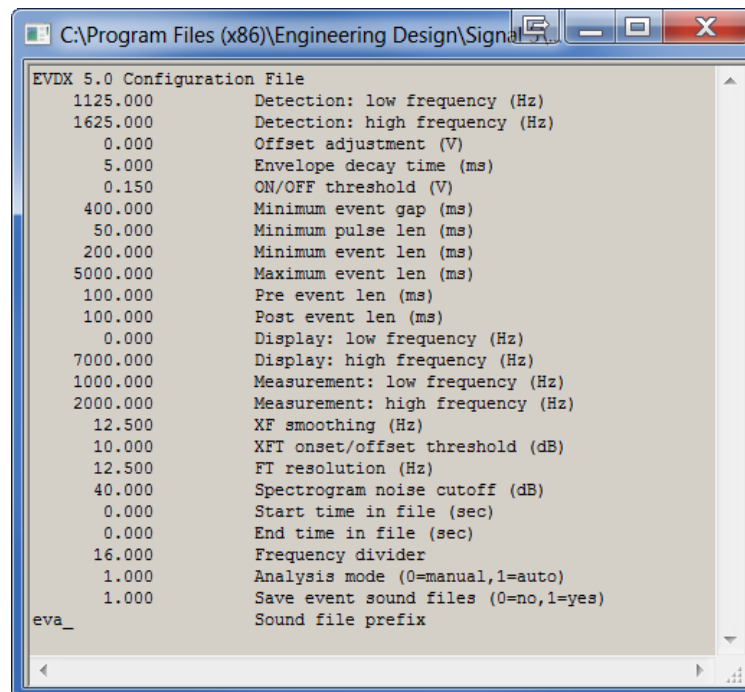
4. Select **Program settings** to open the Program Settings menu. Select **Load settings from config file** from the Program Settings menu to open a list of available configuration files. Select evd\_demo.cfg from the list and click OK.



5. The program will confirm that the settings have been loaded.



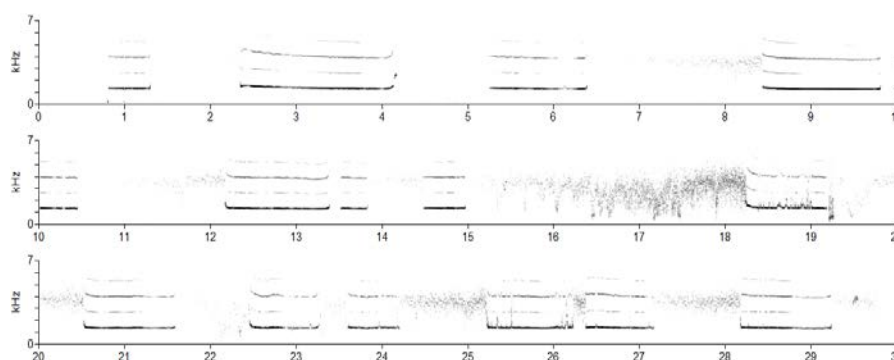
6. Select **Show settings** to display program settings. **Make sure your settings match those settings shown here**, then click OK to proceed.



7. Select **Exit to previous menu** to return to the EVDX main menu.

## Display input sound file

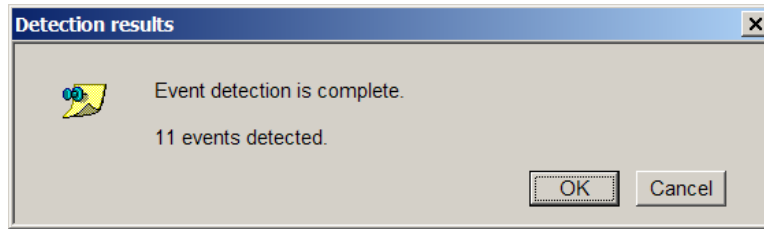
8. Rather than acquiring new sound data, this demo uses the demo sound file **evanal\_demo.sig**, so that demo results can be predicted and confirmed. **evanal\_demo.sig** contains a 30-second rodent vocalization, recorded at 1/16 actual frequency through the frequency-divider circuit of a bat detector. This demo configures EVDX to rescale frequency values stored in the log file accordingly. See "Adjusting Parameters for Frequency-Divided Sound Data" for background.
9. To display the demo sound file, select **Display sound file / detection function** from the EVDX main menu, then select **Spectrogram** for display type. At "Select sound file", enter **evanal\_demo.sig** in the box and click OK. EVDX will draw the entire sound file in 10-second strips, using as many screens as necessary (in this demo, the entire file fits on one screen). Click OK, then select Exit at "Display sound file" to return to the EVDX main menu.



10. EVDX will automatically save every screen as an image files (BMP format) in the **EA user directory** (see Note 2 above). This tutorial has just saved one image file called **evanal\_demo.sig\_000.bmp**. Browse to the EA user directory in Windows Explorer and double-click on this file to display it.

## Perform event detection

11. If you have run this tutorial before, navigate to the EA user directory in Windows Explorer and delete the saved sound files, to avoid answering overwrite queries. These files are named **eva\_0001.sig**, **eva\_0002.sig**, etc.
12. Select **Perform event detection** to begin event detection. At "Select sound file to analyze", enter **evanal\_demo.sig** in the box and click OK. At "Select log file to save", enter **evanal.log** and click OK. Approve overwriting if asked. At "Event sound file directory", browse to <sigdata>\evanal, which is **c:\Engineering Design\Signal 5\userdata\evanal** in Windows Vista and later (see Note 2 above). Click OK to begin detection. Approve overwriting of stored sound files if asked.
13. EVDX will automatically detect all events in **evanal\_demo.sig**, reporting the number of events detected. EVDX should detect 11 events. When detection is complete, click OK to return to the EVDX main menu.



### Examine event log file

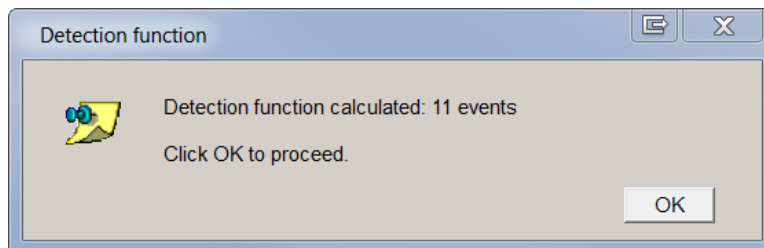
14. Select **Display event log file**. At "Select file", enter **evanal.log** and click OK. Scroll through the log file. Note the sequential event numbering (REC#), increasing start times (TSTART), peak frequency (FPEAK), and other measured parameters. **Your log file should match the log file shown here.** Click OK to close the display window and return to the EVDX menu.

V5.0 EVANAL event log file

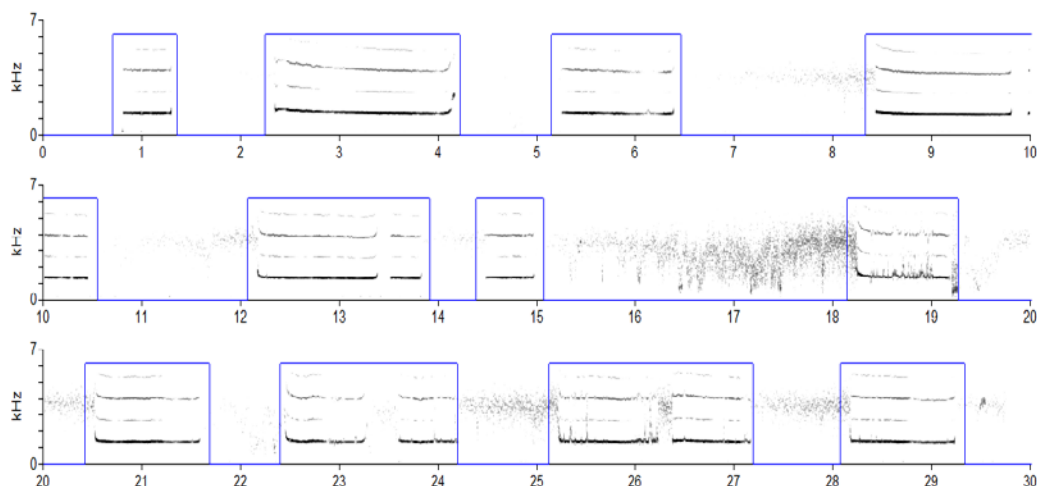
REC#	TSTART	TINTER	DUR	RMS	FPEAK	FMIN	FMAX
1	0.703	0.703	0.648	0.295	21021	20703	21289
2	2.241	0.890	1.979	0.316	20972	20508	24609
3	5.145	0.925	1.315	0.250	20752	20313	21484
4	8.326	1.867	2.221	0.301	19983	19727	22266
5	12.064	1.517	1.854	0.304	20612	20117	21875
6	14.379	0.461	0.689	0.265	20923	20703	21289
7	18.139	3.071	1.122	0.311	21423	20703	25195
8	20.420	1.158	1.260	0.295	21301	21094	23047
9	22.396	0.716	1.801	0.230	21851	20703	24219
10	25.116	0.919	2.071	0.293	21051	20898	22852
11	28.070	0.882	1.261	0.284	21277	20703	22266

### Examine Detection Results On-screen

15. See "Sound File / Detection Function" for additional details and illustrations of the following steps.
16. Select **Calculate detection function**. At "Select log file to analyze", enter **evanal.log**. At "Select file to receive detection function", enter **evanal.det** and approve overwriting if asked. Click OK to return to the EVDX main menu.



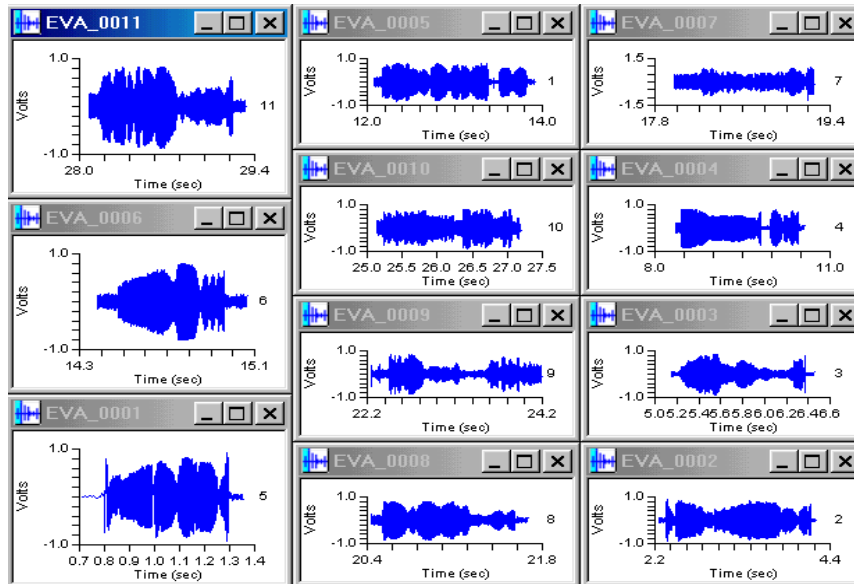
17. Select **Display sound file / detection function** from the EVDX main menu, then select **Spectrogram+detection function** for display type. At "Select sound file", enter **evanal\_demo.sig** and click OK. At "Select detection function file", enter **evanal.det** and click OK.




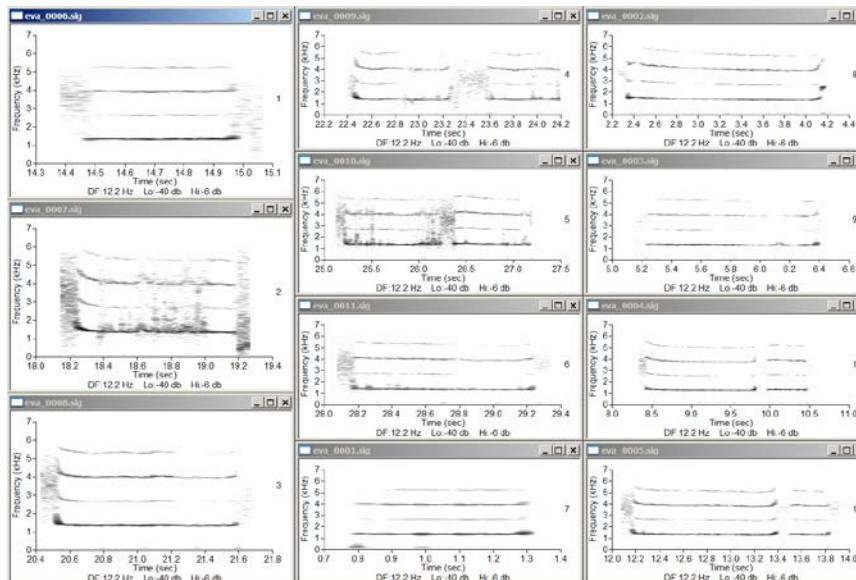
18. Click OK to close the sound file display, then select **Exit** at "Display sound file" menu to return to the EVDX main menu. Then select **Exit Event Detection** to return to Event Analyzer main menu.

### **Examine detected sound files**

19. This tutorial has configured the Event Detection module to save the detected events as sound files, to serve as a dataset for further measurement and analysis by other SIGNAL tools. We'll now exit the Event Analyzer and display the sound files detected in this tutorial.
20. It is convenient to drag a large number of sound files into SIGNAL for viewing, rather than opening them individually via File | Open. Security rules in Windows Vista and later prevent dragging from Windows Explorer, but an Explorer substitute can be used. **Explorer++** is recommended. For details, see "Drag and drop / User Account Control" in the **SIGNAL 5 Installation Guide** at <http://www.engdes.com/sigwin/support/>
21. Follow these steps:
  - 1) Select **Exit Event Analyzer** to exit the Event Analyzer.
  - 2) Select Tools|Delete Buffers|All to clear all buffers.
  - 3) Select Tools|Preferences|General on the menu and make sure "Show file open dialog" is **not** checked.
  - 4) In Windows Explorer or Explorer++, navigate to the Event Analyzer user directory <sigdata>\evanal (see "Directory Structure" above), which should contain 11 files named **eva\_0001**, **eva\_0002**, etc.
  - 5) Drag these files into SIGNAL or use File | Open (see the note above).
  - 6) In SIGNAL, select Window|Tile Horizontally to display the 11 waveforms. The result should resemble the following, though the order may differ.



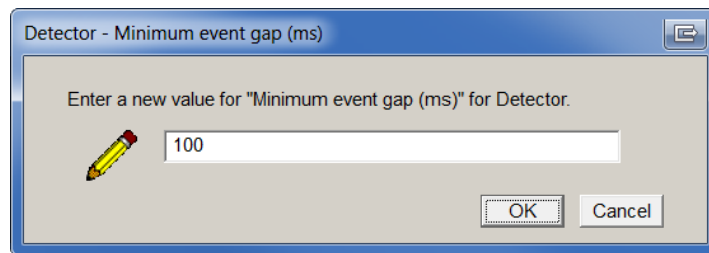
- 7) Any signal in the display can be converted to a spectrogram by selecting its window and clicking the spectrogram toolbar button  and specifying a destination buffer for the spectrogram. Set the destination to the number shown to the right of the waveform in each graph window, so each spectrogram is stored in a unique spectrogram buffer.



22. Select Tools|Delete Buffers|All in the SIGNAL menu to clear all buffers.

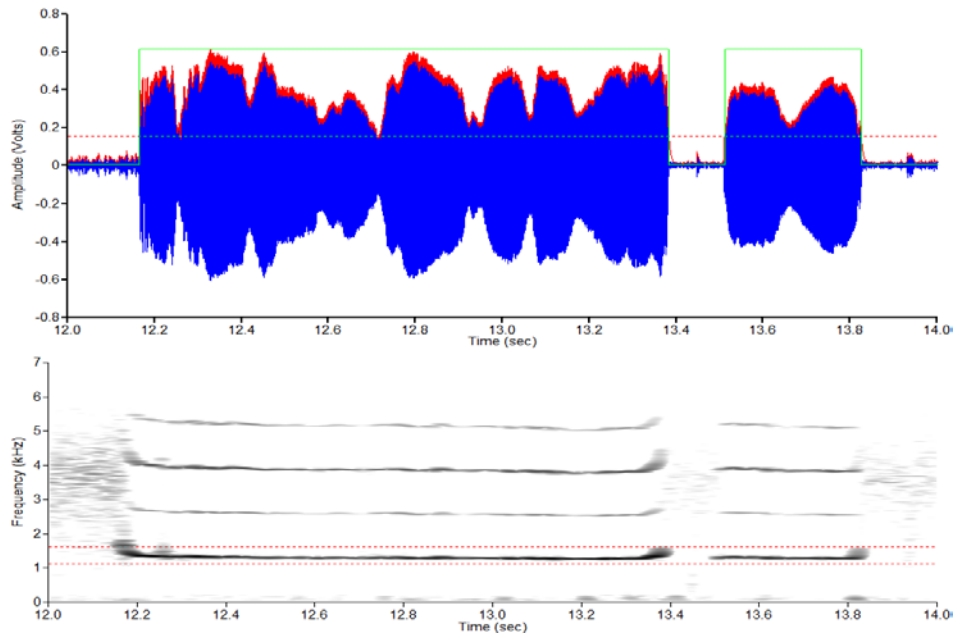
### **Detection model**

23. The EVDX **detection model (DM)** allows the user to diagnose and refine the detection process interactively on-screen. The user can alter detection parameters, recalculate detection functions, and observe the effect. See "EVDX: Detection Model" for details, theory and illustrations of the following steps.
24. Perform steps 2 and 3 to start EVDX again.
25. Select **Program settings** to open the Program Settings menu. Select **Load settings from config file** from the Program Settings menu to open a list of available configuration files. Select **evd\_demo.cfg** from the list and click OK.
26. In **Program settings**, select **Change settings**, then select **Detector event lengths**. Click OK until you get to **Minimum event gap**, then change that setting to 100 msec. This value will create a detection issue that we will observe and then correct.

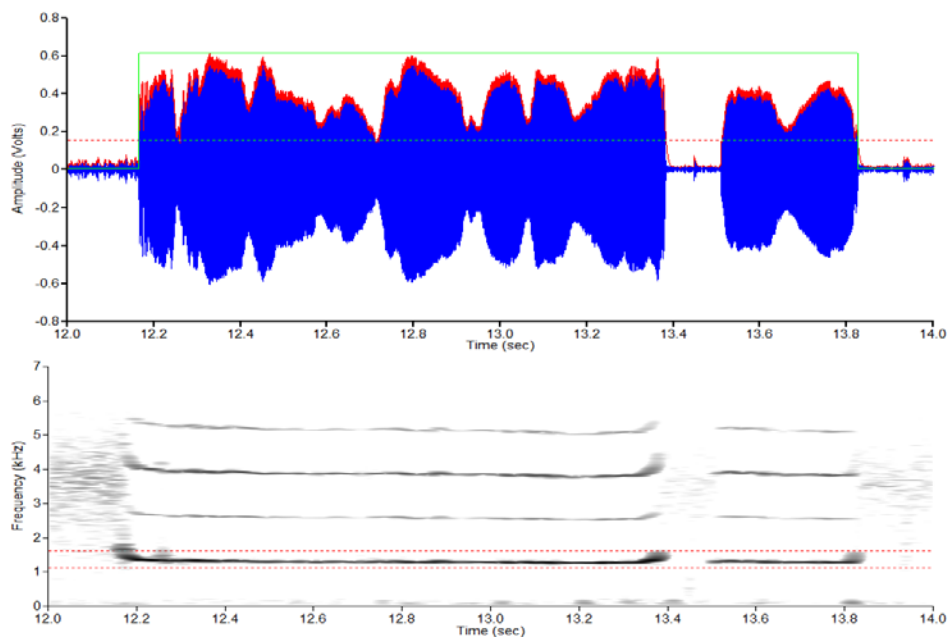


27. Click OK for each of the remaining settings to accept their current values without change. At the **Change program parameters** menu, select **Exit to Program Settings menu**, then **Exit to previous menu** to return to the EVDX main menu.
28. Select **Analyze detection parameters**, then at "Select sound file", enter **evanal\_demo.sig** and click OK.
29. At "Start time" enter 12 and at "End time" enter 14, to set the source file interval to analyze.

30. The resulting screen display shows the waveform (blue), amplitude envelope (red) and detection function (green).



31. The two green overhead boxes indicate that the detection function has parsed the two pulses as two separate events. This is because **minimum event gap** is too small, causing the **intra-event space** to be interpreted as an **inter-event separator**.
32. Select **Change minimum event gap** from the **Adjust detection parameters** menu and enter 400. The two pulses are now be detected as one event (a single green overhead box).

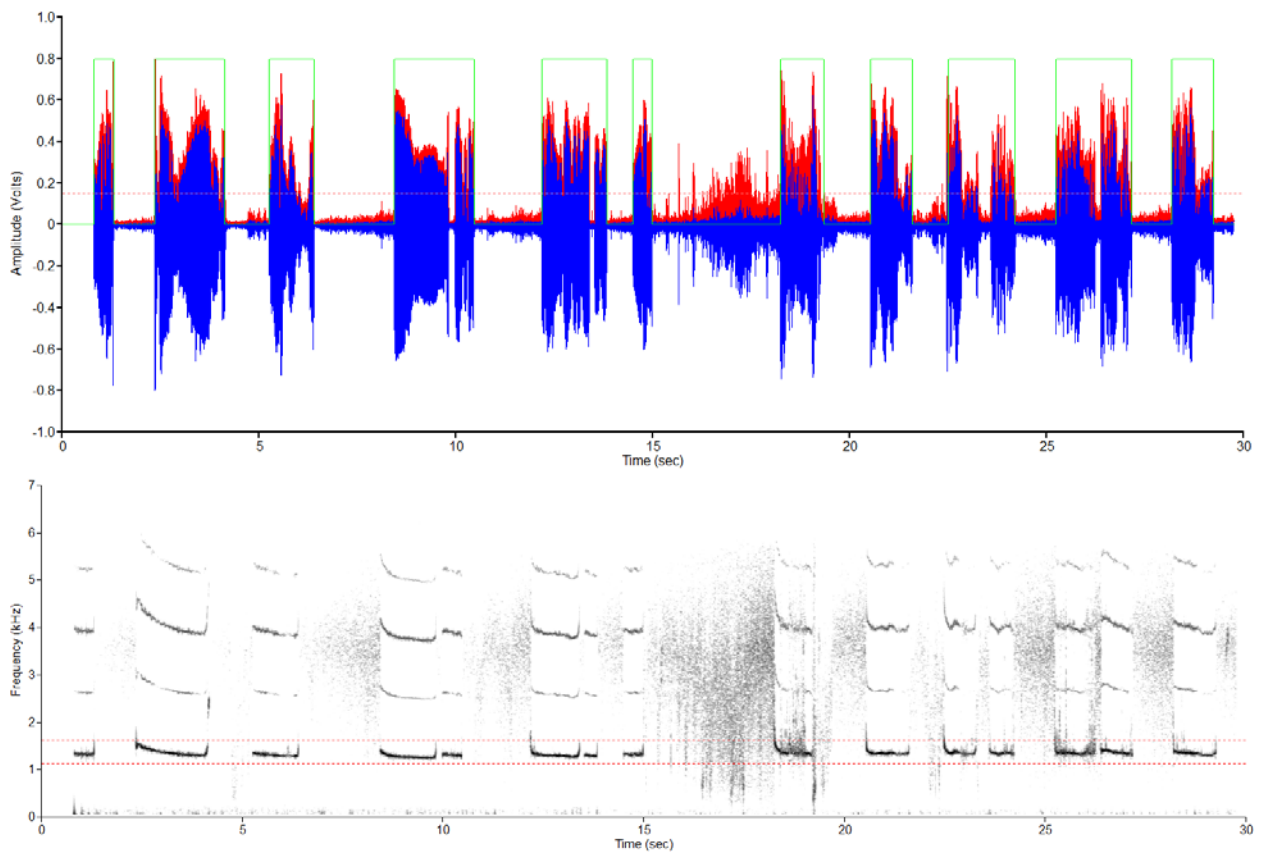


33. For example, with birdsong, the Event Analyzer can parse and deliver individual notes or it can aggregate multiple notes into one song and deliver individual songs. This is

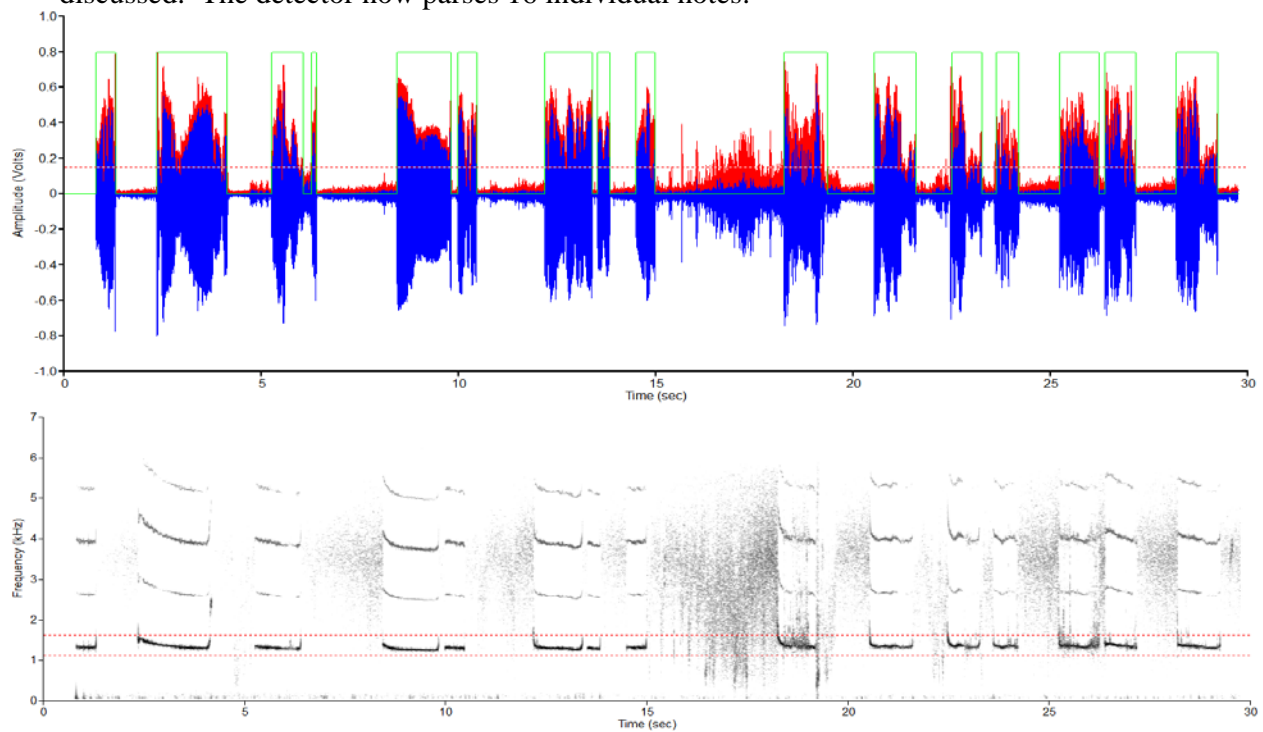


controlled by the detection parameters, principally **minimum event gap** and **ON/OFF threshold**. In the previous example, imagine that notes generally lie within 200 msec of each other within a song, while songs are separated by at least 500 msec. Then a **minimum event gap** of 400 msec will ensure that 1) individual notes are connected into one song, because the max intra-song note separation of 200 msec < 400 msec cutoff, as we have just seen and 2) individual songs are parsed as separate events (such as the 11 events detected in this tutorial), because inter-song separation > 400 msec cutoff.

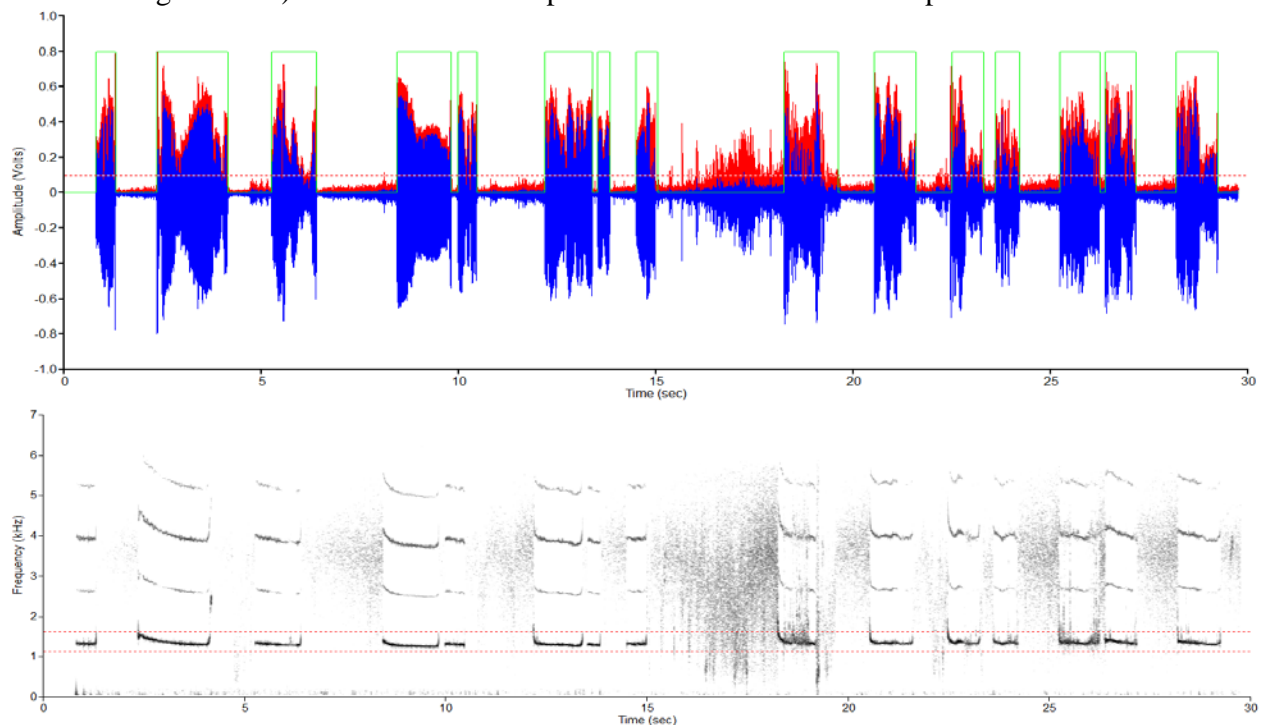
34. Let's illustrate this concept by adjusting the detection parameters to parse the same sound sample into individual notes.
35. Select **Save changes and exit** to return to the EVDX main menu. Select **Analyze detection parameters**, then at "Select sound file", enter **evanal\_demo.sig** and click OK. At "Start time", keep the default of 0 and at "End time" keep the default of 29.75.
36. The green overhead boxes in the resulting screen display show the 11 "song" events that we have already detected in this sound sample.



37. Let's adjust the detection parameters to detect individual notes. Select **minimum event gap** and enter 100 to make the detector sensitive to the gaps between notes, as just discussed. The detector now parses 16 individual notes:



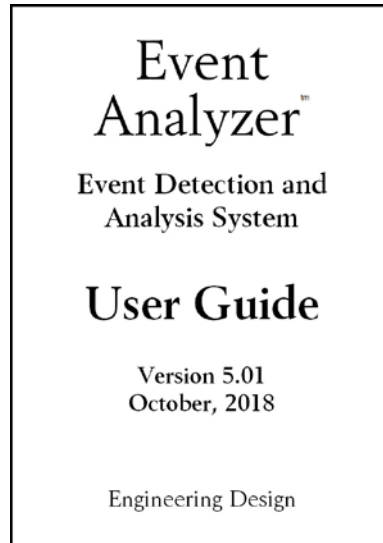
38. The third song may consist of only one note. We can correct this by lowering the detection threshold to bind those two notes together. Select **Change detection threshold**, move the crosshair cursor into the **upper plot** to  $Y=0.10$  (use the X-Y readout in the lower right corner) and hit <esc> to keep this value. The detector now parses 15 notes:



39. Select **Discard changes and exit** to return to the EVDX main menu.

#### **Miscellaneous commands**

40. Select **Help**. Select "Event Analyzer User Guide" and click OK. Wait while the guide loads into Adobe Acrobat. At "Opening Event Analyzer User Guide", click OK, then select **Exit to previous menu** and click OK again to return to the EVDX main menu. The guide may be left open.



#### **Exit from tutorial**

41. Select **Program settings**, then **Load settings from config file**, then **evd\_default.cfg** from the file list to restore default detection settings. Then select **Exit to previous menu** to return to the EVDX main menu.
42. Select **Exit Event Detection** and **Exit from Event Analyzer** to exit the Event Analyzer.

## **14. Event Analysis (EVAX) Overview**

---

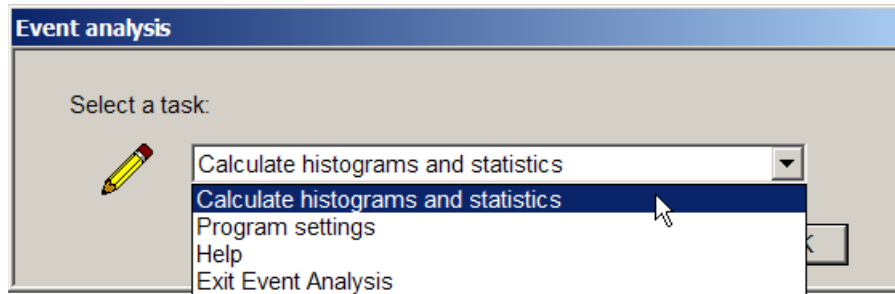
The **event analysis** module (also referred to in this guide as "EVAX") performs statistical analyses on the event log file produced by the **event detection** module. These analyses can be individually configured then selected for screen display. EVAX analyses include event duration, interevent interval, peak frequency, summary statistics such as call rate, average event duration, mean peak frequency, and histogram distributions of these parameters.

This section provides an overview of event analysis. Detailed operations are discussed in subsequent sections.

### **Event Analysis Menu**

---

Following is the event analysis menu:



---

## Calculate histograms and statistics

This menu selection calculates summary statistics such as call rate, average event duration, mean peak frequency, and displays histogram distributions of these parameters. EVAX produces no data files – all results are displayed to the screen.

---

## Program settings

Event analysis is configured by a number of parameters, referred to as **program settings**. This selection opens the Program Settings menu, which allows the user to view, change, load or save program settings.

---

## Help

Display the Event Analyzer User Guide (this document).

---

## Exit Event Analysis

Exit the Event Analysis module, back to the main Event Analyzer menu.

---

# 15. EVAX: Program Settings

---

Event analysis is configured by a number of parameters, referred to as **program settings**.

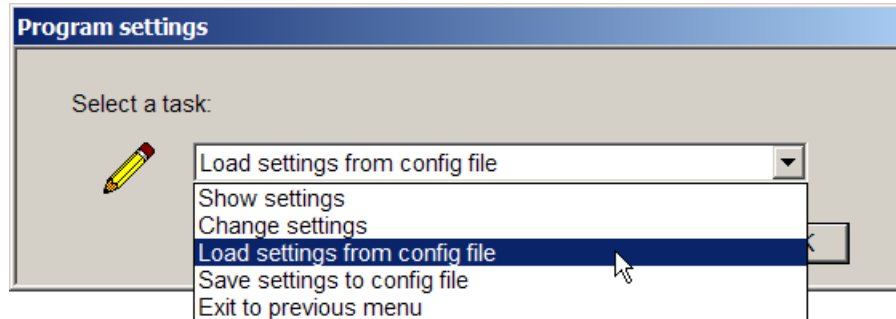
---

## Configuration Files

See "Configuration files" under "EVDX" Program Settings".

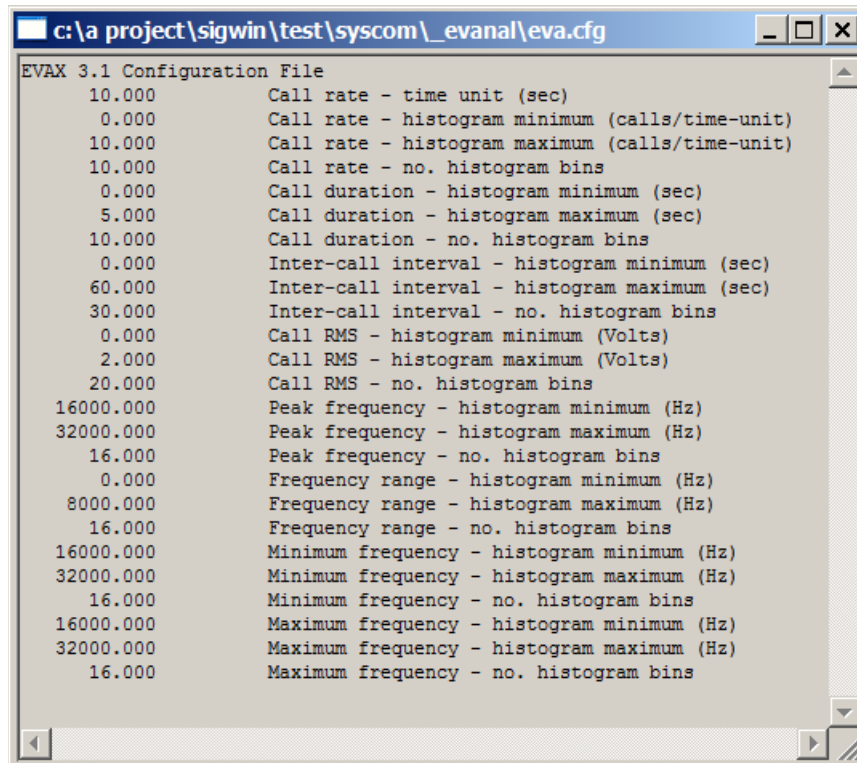
## Program Settings Menu

The Program Settings menu allows the user to display, change, load or save current program settings.



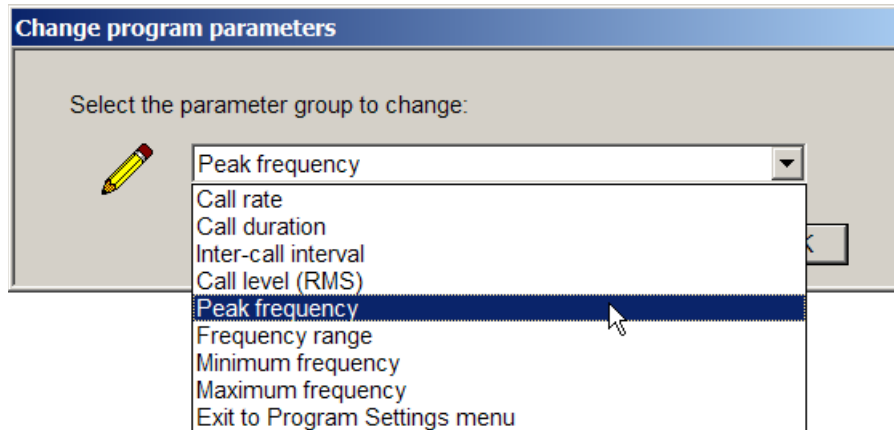
## Show Settings

This selection displays the current program settings, stored in the working event analysis configuration file eva.cfg.

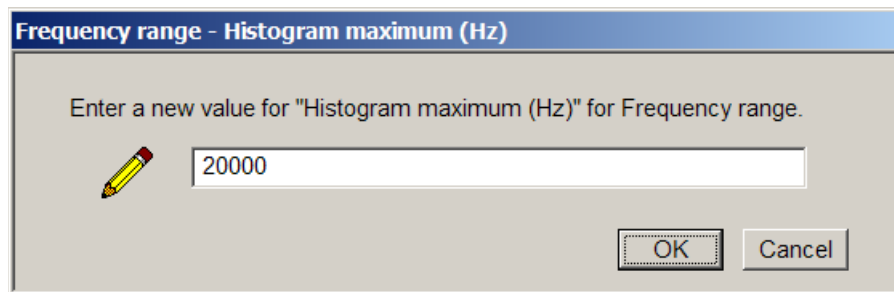


## Change Settings

This selection allows the user to alter individual parameter values. It displays a menu of parameter groups:

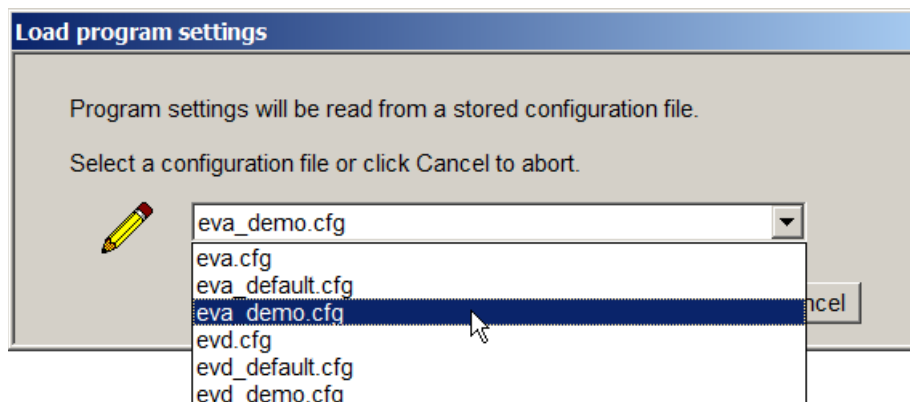


The user selects a group, then the event detector displays the parameters in that group, one by one, with their current value. The user can enter a new value or hit <enter> to leave the parameter unchanged. In the following example, the user changes maximum histogram limit for the Frequency range display to 20000 Hz:



## Load Settings from File

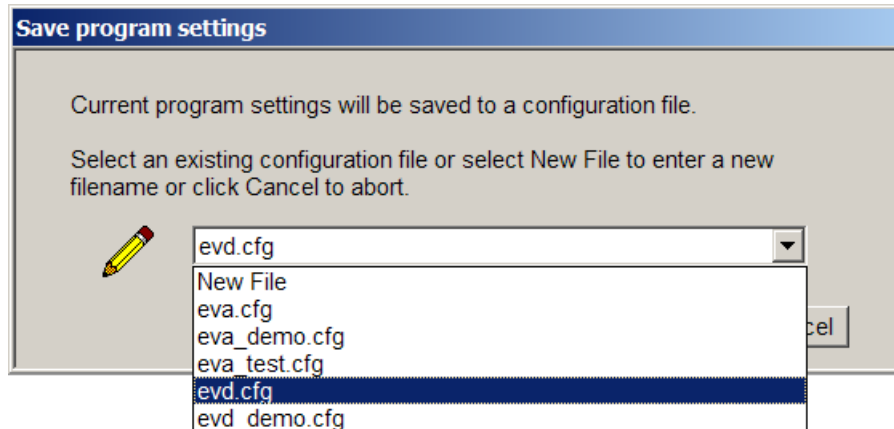
This selection loads current program settings from a file specified by the user. Note that EVDX and EVAX maintain separate configuration files, and both file types will appear in the file list (see the figure). **Be sure to load the correct configuration file type** – or unpredictable behavior may result.



## Save Settings to File

---

This selection saves current program settings to a specified configuration file. Select an existing file or select "New File" and enter a filename when prompted. The filename must end with a ".cfg" suffix so the program will recognize it.



## Description of Parameters

---

Following is a description of EVAX configuration parameters. The "EVDX: Event Log File" section discusses the calculation of many of the quantities displayed by EVAX.

**Note** that if the source file is frequency-divided (see "Adjusting Parameters for Frequency-Divided Sound Data" under EVDX above), **all frequency parameters** (such as peak frequency, frequency range, minimum frequency, etc.) should be divided by the frequency divider factor.

### Call Rate

Call rate is the number of calls detected within a specified time unit, such as one minute. To derive this statistic, the data set is partitioned according to this time unit and the number of calls in each unit is counted. EVAX displays a histogram of these count values and reports their mean value. **Call rate - time unit** sets the time unit. **Histogram minimum, histogram maximum, and no. histogram bins** set histogram limits and number of bins.

### Call Duration

Call duration is the duration in seconds of each event. EVAX displays a histogram of call durations and reports the mean value. **Histogram minimum, histogram maximum, and no. histogram bins** set histogram limits and number of bins.

### Inter-Call Interval

Inter-call interval is the time in seconds between the start of the current call and the end of the previous call. EVAX displays a histogram of inter-call intervals and reports the mean

value. **Histogram minimum, histogram maximum, and no. histogram bins** – set histogram limits and number of bins.

### **Call RMS**

Call RMS is the RMS (root-mean-square) level in Volts of each call. EVAX displays a histogram of RMS call levels and reports the mean value. **Histogram minimum, histogram maximum, and no. histogram bins** set histogram limits and number of bins.

### **Peak Frequency**

Peak frequency is the frequency value in Hz at which the power spectrum of the event achieves its maximum value. EVAX display a histogram of peak frequencies and reports the mean value. **Histogram minimum, histogram maximum, and no. histogram bins** set histogram limits and number of bins.

### **Frequency Range**

Frequency range is the difference in Hz between the minimum and maximum frequencies present above a specified level in the power spectrum of the event. EVAX displays a histogram of frequency ranges and reports the mean value. **Histogram minimum, histogram maximum, and no. histogram bins** set histogram limits and number of bins.

### **Minimum and Maximum Frequency**

Minimum and maximum frequency are the lowest and highest frequency values in Hz that are present in the detected event, within a specified frequency bandwidth, and above a specified minimum intensity level. These parameters are measured from the event spectrogram. EVAX displays a histogram of minimum or maximum frequencies and reports the mean value. **Histogram minimum, histogram maximum, and no. histogram bins** set histogram limits and number of bins.

## **16. EVAX: Output Displays**

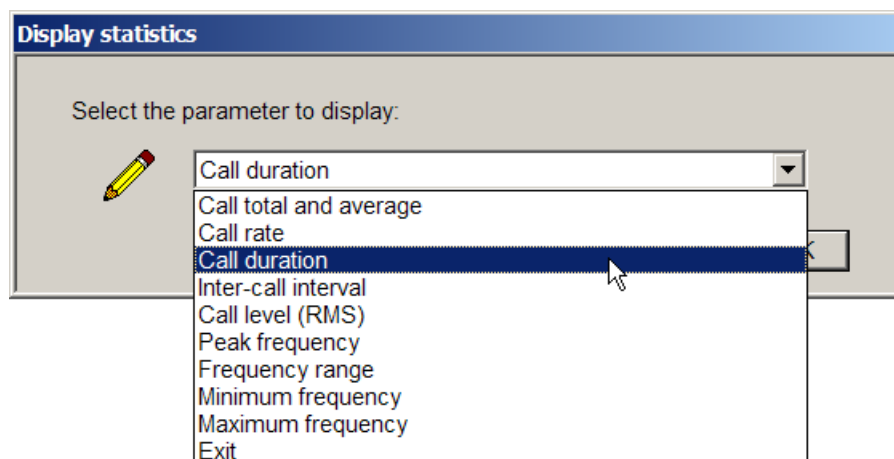
---

### **Display Menu**

---

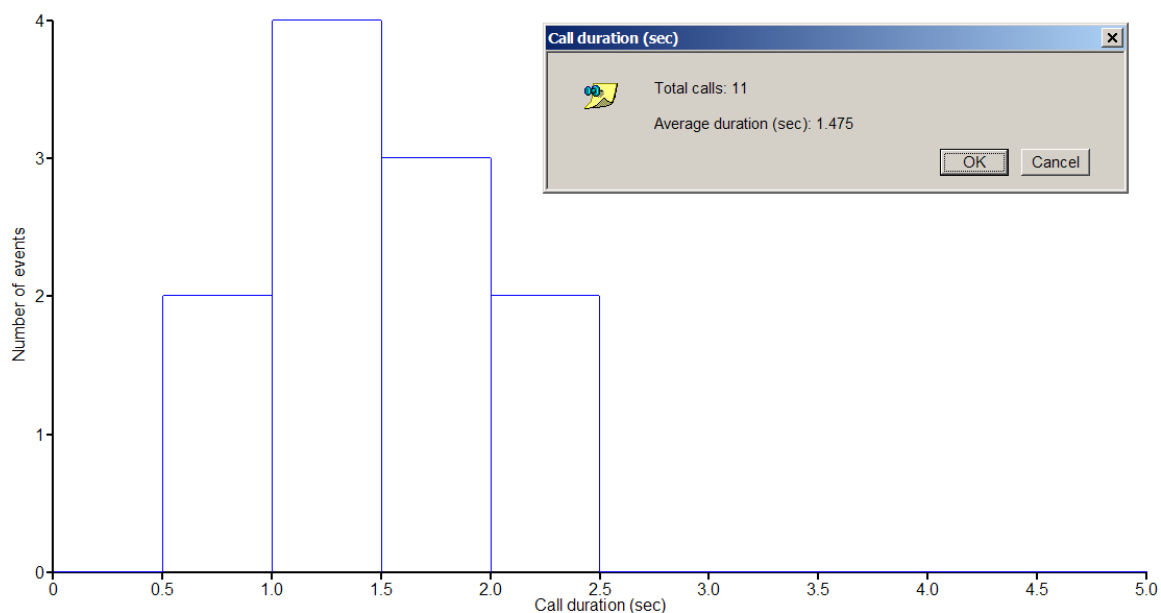
To view output displays, select "Calculate & display histograms and statistics" from the EVAX menu. EVAX will query for an event log file, analyze that file statistically, then present the Display statistics menu:





## Display Output

Select a variable from the menu. EVAX will then display a histogram distribution plus statistical summary, typically the total events analyzed and the mean value of the selected variable. For example, the following figure shows the histogram and summary for Call duration.



## 17. EVAX: Tutorial

The following tutorial analyzes the event log file created in the section "EVDX: Tutorial" and provides histogram displays and statistics.

**Note 1:** Be sure to review "Directory Structure" in this guide before running this tutorial, to understand where Event Analyzer components are stored.

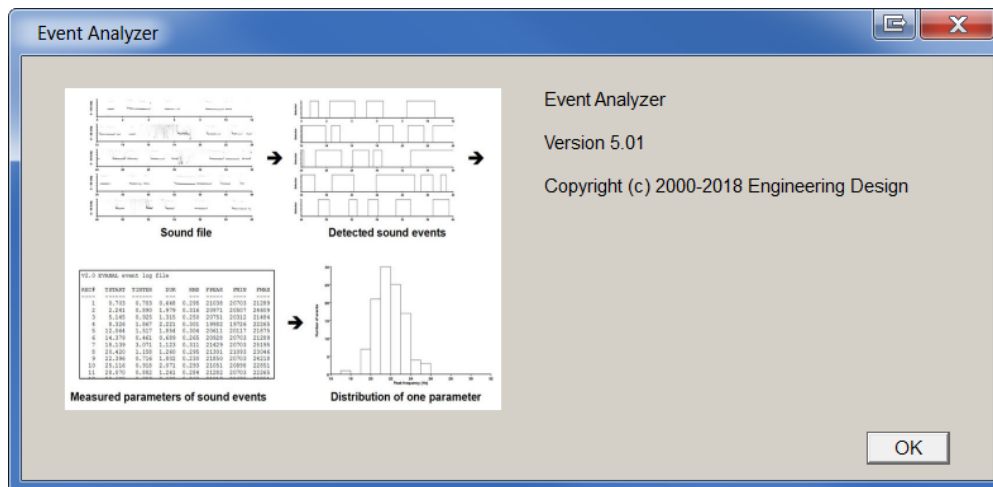
**Note 2:** If desired, you can examine output files while running through the tutorial:

- Output data – including the log file, image files, and event sound files – are stored in the **EA user directory**, which is **c:\Engineering Design\Signal 5\userdata\evanal** in Windows Vista and later (see "Directory Structure").
- Event Detection reference files – such as demo signals and the reference demo log file – reside in the **EA system directory**, which is **c:\Program Files (x86)\Engineering Design\Signal 5\syscom\evanal** in Windows Vista and later (see "Directory Structure").

**Note 3:** If necessary, you can always restore default EVAX program settings as follows. Select "Event analysis" at the main Event Analyzer (EA) menu, select **Program settings**, then **Load settings from config file**, then "eva\_default.cfg" from the file list.

### Launch the Event Analyzer

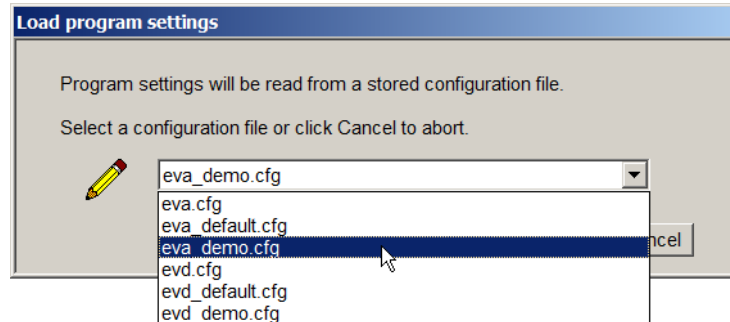
1. Start SIGNAL.
2. The EVAX tutorial requires the output file **evanal.log** containing the specific measurements stored by the EVDX tutorial. If you have not yet run the EVDX tutorial, go through the "EVDX: Tutorial" to create this file.
3. Select Signal | Event Analyzer from the SIGNAL menu or type "evanal" in the console window to launch the Event Analyzer.



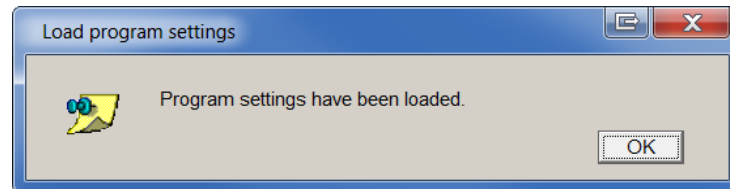
4. Select "Event analysis" from the Event Analyzer menu.

### Set parameters for Event Analysis

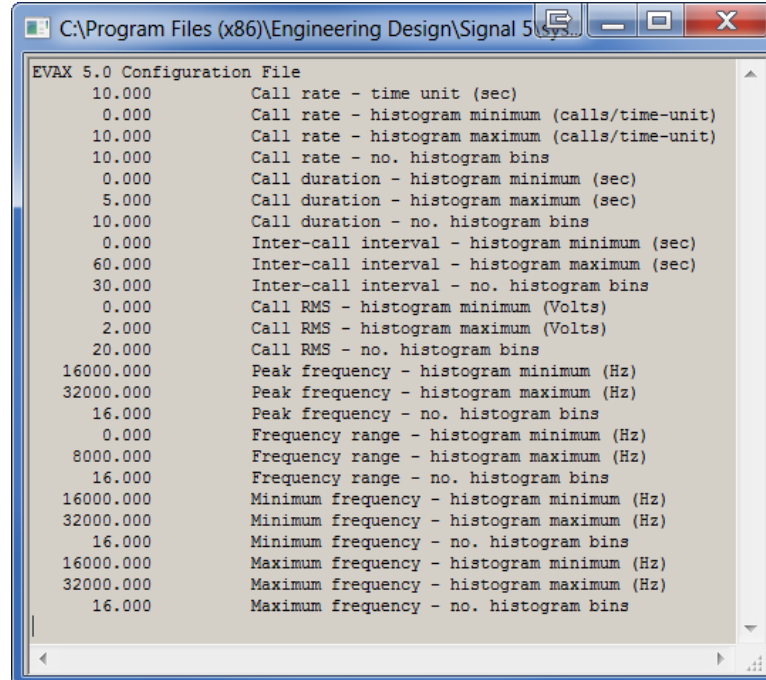
5. Select **Program settings** to open the Program Settings menu. Select **Load settings from config file** from the Program Settings menu to open a list of available configuration files. Select **eva\_demo.cfg** from the list and click OK.



6. The program will confirm that the settings have been loaded.



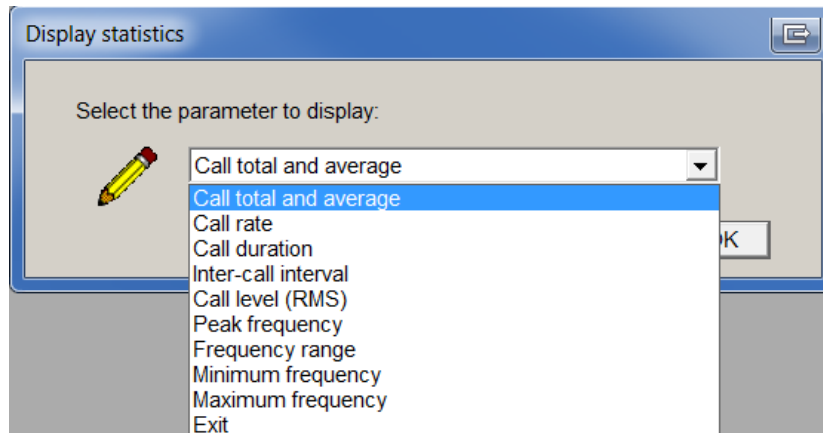
7. Select **Show settings** to display program settings. **Make sure your settings match those shown here**, then click OK to proceed.



8. Select **Exit to previous menu** to return to the EVAX main menu.

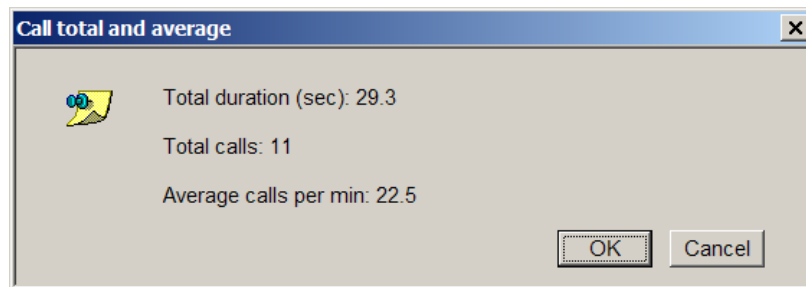
### Perform event analysis

9. Select **Calculate histograms and statistics** from the menu. At "Select log file to analyze", enter **evanal.log**, then click OK to begin processing.
10. EVAX will automatically analyze all events in **evanal.log**, then show the **Display statistics** menu.

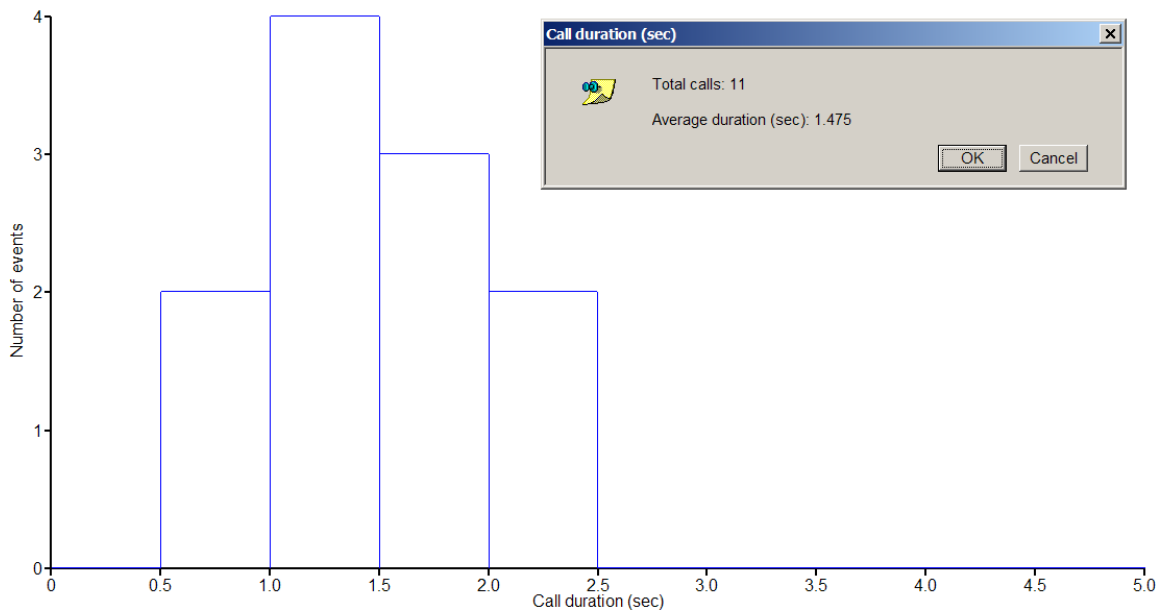


### Examine analysis output

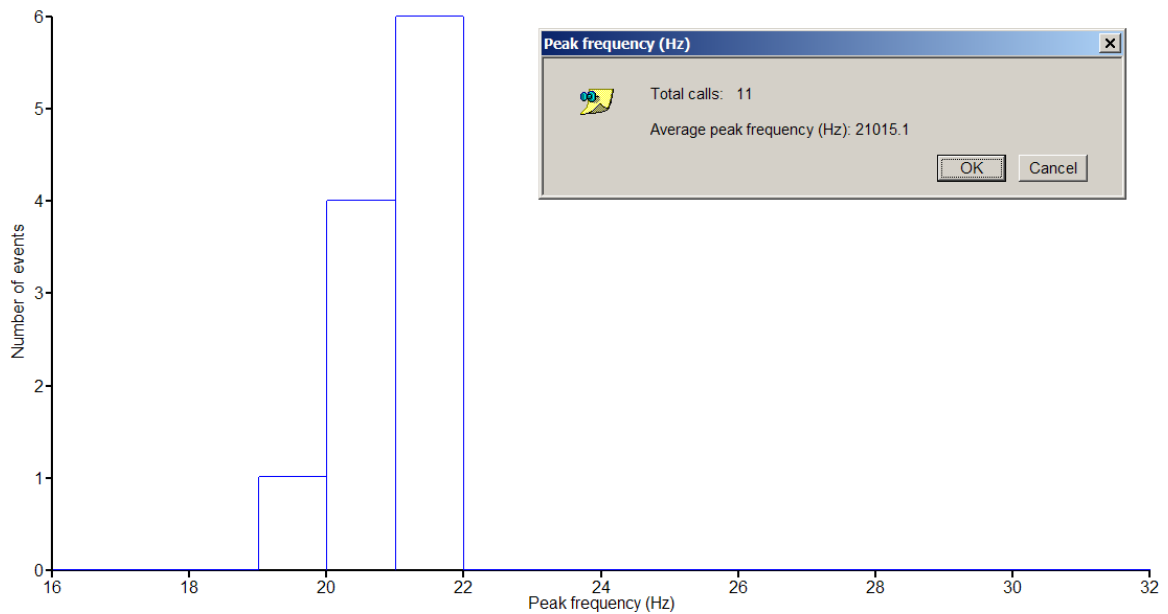
11. Select **Call total and average**. EVAX should report the duration of the original sound file, number of calls detected, and average calls per minute. Click OK to return to Display statistics menu.



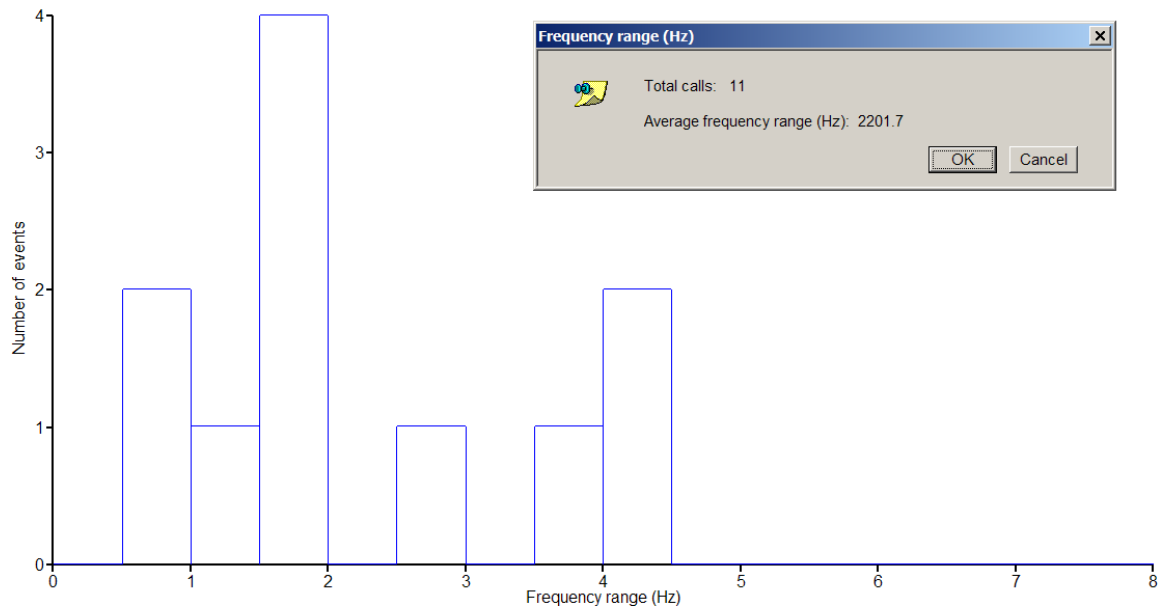
12. Select **Call duration** to view a histogram of the 11 call durations and the average call duration (1.475 sec), then click OK.



13. Select **Peak frequency** to view a histogram of the 11 peak frequency measurements and the average peak frequency (21015.1 Hz), then click OK.



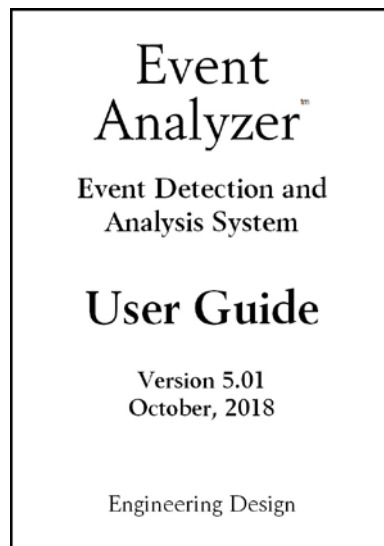
14. Select **Frequency range** to view a histogram of the 11 frequency range measurements and the average frequency range (2201.7 Hz), then click OK.



15. Select **Exit** to exit the Display statistics menu and return to the EVAX main menu.

### Miscellaneous commands

16. Select **Help**. Select "Event Analyzer User Guide" and click OK. Wait while the guide loads into Adobe Acrobat. At "Opening Event Analyzer User Guide", click OK, then select **Exit to previous menu** and click OK again to return to the EVAX main menu. The guide may be left open.



### **Exit from tutorial**

17. Select **Program settings**, then **Load settings from config file**, then **eva\_default.cfg** from the file list to restore default detection settings. Then select **Exit to previous menu** to return to the EVAX main menu.
18. Select **Exit Event Analysis** and **Exit from Event Analyzer** to exit the Event Analyzer.

## **18. Operating Hints**

---

Following are operating hints for EVDX and EVAX.

1. The STRIP command in SIGNAL is useful for displaying long signals. STRIP can display a continuous waveform or spectrogram in strip-chart format on multiple screens. See the **SIGNAL Reference Guide** for details.
2. EVDX can accept the frequency-divided sound data provided by bat detectors, however frequency configuration parameters must be scaled accordingly. See "Adjusting Parameters for Frequency-Divided Sound Data" in "EVDX: Program Settings".
3. EVDX detection parameters are partially interdependent. For example, increasing envelope decay time will extend apparent signal lengths and thereby effectively lower the detection threshold. Reducing the bandpass filter range may reduce the received signal level, thereby raising the effective detection threshold.
4. EVDX specifies all event detection parameters (such as signal lengths, event gap, etc.) in msec. All other time parameters in EVDX and EVAX are specified in seconds.
5. The measurement of minimum and maximum frequency in EVDX is very sensitive to the XFT onset/offset threshold parameter. Increasing the parameter value (which lowers the threshold) will generally increase the detected frequency range, but at the risk of detecting noise energy instead of the actual signal.