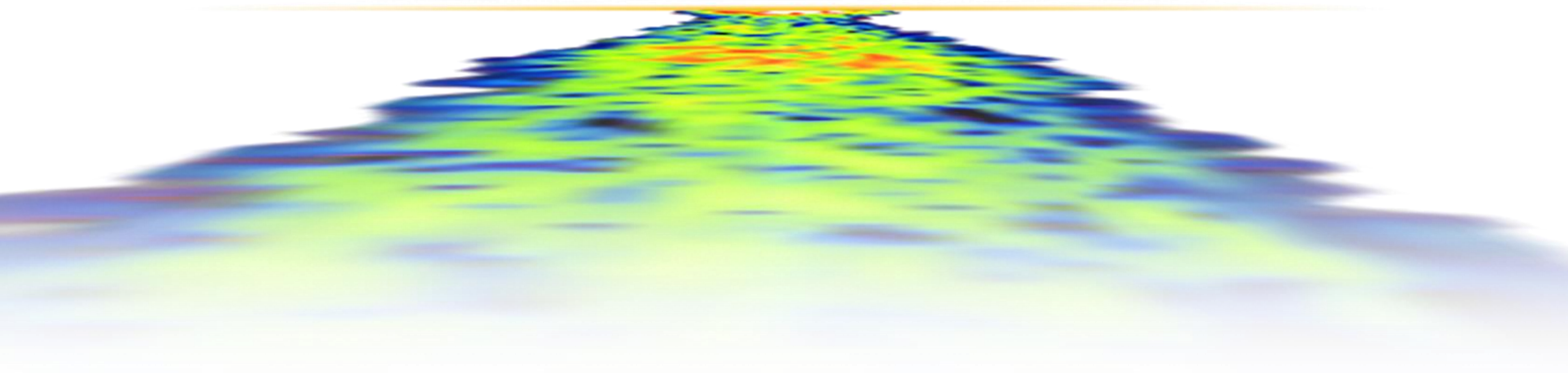
Smaart® v8

rational acoustics



User Guide

rational acoustics

Rational Acoustics

Smaart v8

User Guide

Release 8.3

rational acoustics

Rational Acoustics Smaart v8 User Guide Copyright notice

For information contact:

Rational Acoustics, LLC
32 Crabtree Lane
Woodstock, CT 06281 USA

telephone: (+1) 860 928-7828
e-mail: info@rationalacoustics.com
web: http://www.rationalacoustics.com

Contents

Table of Figures

Introduction

What is Smaart?

Rational Acoustics Smaart® is a dual-channel, FFT-based acoustical analysis software application that runs on Microsoft Windows and Mac OS X. It provides real-time spectrum analysis of audio signals, dual-channel transfer function analysis of sound system response and acoustical impulse response measurement and analysis capability. Smaart enables you to measure and analyze the frequency content of audio signals, study the timing and frequency response of electro-acoustic systems, and perform basic room acoustics analysis.

Smaart is designed to be accessible to a wide cross-section of audio engineers and technicians, offering the flexibility and scalability to meet the requirements for nearly any field measurement and analysis application while maintaining a level of intuitive usability that everyone from novice users to the busiest seasoned professionals can appreciate. Being entirely software-based, Smaart is hardware independent and can process data from nearly any audio source that can stream data to a computer, from built-in sound chips in laptop and tablet computers, to professional multi-channel recording interfaces and digital mixing consoles, to complex networked digital audio systems.

Scope and Purpose of this Guide

This guide is intended as a practical introduction to configuring and operating Smaart v8. Our goals are to provide a comprehensive explanation of the program and its features and operation along with a brief survey of some of the core concepts related to acoustic measurement and analysis, and to establish a foundation for making valid, repeatable measurements and extracting some useful information from the results. This is not a book on sound system engineering or acoustical measurement in general and we would strongly urge anyone new to the subject who is serious about learning it to go and read one, or perhaps several. A list of some additional sources of information is provided in the appendices.

We assume that the reader has a basic understanding of professional audio equipment and engineering practices. A separate guide entitled *Choosing gear for your Smaart measurement system* is available from our web site that discusses the basics of computer audio I-O devices and hardware related specifically to acoustical measurement and analysis, such as measurement microphones and sound level calibrators.

Regardless of past experience with previous versions of Smaart, or other measurement and analysis systems, you will need to take the time to familiarize yourself with configuring and operating Smaart version 8. Smaart offers *extensive* flexibility in terms of the number of inputs you can analyze and the number of ways you can display the results. It can interface with multiple I/O devices simultaneously and run multiple real-time displays in multiple windows, each with multiple workspaces set up on tabbed pages. Upon first run, however, Smaart begins with a simple RTA graph and no preconfigured measurement setups. It is up to the operator to take it from there, and configure a work environment that makes sense their specific applications.

How to use this Guide

This guide is organized in such a way that it *can* be read from start to finish. We have tried to present information about the program and its various features and options organically, and in context. As a result, if you need to find details about a single specific button or feature, it may not be in the first place that you might think to look. We have provided an extensive table of contents and an index to help readers track things down by topic, and of course if you are reading an electronic copy you can do a full text search. Alternatively, you can use Smaart's online help system, which contains much of the same information but is organized more in parallel with the user interface of the program, to "drill down" through menus, dialogs, and on-screen controls to find what you are looking for.

Notation for Accelerator Keys (Hot Keys) and Mouse Clicks

Smaart runs on both Windows and Mac OS X, meaning that there are some minor differences in keyboard and mouse commands between the two versions. Specifically, the Control [Ctrl] key serves the same purposes on a Windows computer as the Command [Cmd] key (also commonly called the Apple key or flower key) on a Mac. Similarly, the [Alt] key in the Windows version of Smaart maps to the [Option] key on Mac keyboards. Additionally, most PC mice have at least two buttons (left and right) whereas many Mac's have only one.

In this document, we will write the names of keys used for keyboard shortcuts (also called "hot keys" or accelerator keys) in square brackets to distinguish them from other text. In cases where a key has one name on a Windows keyboard and another on a Mac, both names will appear inside the brackets with a slash in between, for example, [Ctrl/Cmd] means press the [Ctrl] key on a Windows machine or the [Cmd] key on a Mac.

Summary of notational conventions for keyboard and mouse operations

Key names for keyboard commands appear in square brackets ([Key Name])

[Ctrl/Cmd] means press the [Ctrl] key on a Windows machine or the [Cmd] key on a Mac.

[Alt/Option] means press the [Alt] key on Windows or the [Option] key on Mac.

Left-click on a Windows machine is a regular mouse click on Mac.

A right-click for Windows users means [Ctrl] + mouse click on Mac.

As regards the mouse (or other pointing device), the left button on a Windows mouse corresponds to a normal mouse click on Mac, so if we say "left-click," Mac users just click and if we just say "click," Windows users left-click. A right-click operation on Windows can be accomplished on a Mac by holding down the [Ctrl] key (not to be confused with the [Cmd] key) while you click. On a touchscreen device, "left-click" may equate to a quick tap on the screen with your finger or stylus and "right-click" may mean a longer press and hold.

"K" versus "k"

A perennial source of ambiguity in literature and documentation regarding DSP hardware and software has been the use of the abbreviation "k" (for kilo) to mean both multiples of 1000 and 1024 (2^{10}). In this guide, we will strive to adhere to the SI convention of using the lower case "k" to denote <u>only</u> multiples of 1000. We will use an upper case "K" when we are talking about multiples of 1024. For example, you should always be able to read 48k as 48000 and 8K as meaning 8192 (8 x 1024).

Properly speaking, 2^{10} probably *should* be abbreviated "Ki" (short for "Kilobinary"), to disambiguate it from "K" for Kelvin or Karat, however there isn't much danger of anyone thinking we measure FFT sizes by weight or temperature, and "Ki" has yet to come into very common usage.

Full Scale (dB FS) versus Full Scale

There exist two competing references for decibels in digital audio signals. One convention references dB FS to the largest positive and negative amplitude values obtainable from a given integer sample word size – e.g., ± 32768 for 16 bits – normalized to a range of ± 1.0, such that 0 dB FS denotes the maximum possible digital amplitude value. We will refer to this as "normalized Full Scale."

The second convention, preferred by the Audio Engineering Society (AES), references 0 dB FS to the *RMS* value of a full-scale peak-to-peak sinewave (i.e., 0.7071 normalized Full Scale, rather than 1.0). We will call this "AES Full Scale". In Smaart, Full Scale decibel values are always referenced to normalized Full Scale, meaning that the RMS magnitude of a full-scale digital sinewave is -3.01 dB FS.

Recommended Computer Hardware

While Smaart v8 will operate on a wide range of computer hardware configurations, we recommend the following minimum computer configuration for new installations:

Windows®

- Operating System: Windows 7 or newer (32 & 64 bit)
- CPU: 2 GHz Dual-Core Intel i5 Processor or faster
- RAM: 2 GB or greater
- Graphics: Intel HD 4000 or better, or 256 MB dedicated video RAM
- Display: Min. 1024 × 768 pixel display
- Sound: Audio Hardware with OS compatible ASIO, Wav/WDM drivers

Macintosh

- Operating System: Mac OS X 10.7 (32 + 64 bit) or newer
- CPU: 2 GHz Dual-Core Intel i5 or faster
- RAM: 2 GB or greater
- Graphics: Intel HD 4000 or better, or 256 MB dedicated video RAM.
- Sound Hardware: Audio Hardware with compatible Core Audio device drivers

Loading and Licensing the Software

To install Smaart v8 on a Windows computer, download and run the Windows installer program and follow the prompts in the install wizard. On a Macintosh computer, open the installation disk image and drag the Smaart application bundle into your applications folder.

Registering your installation

Following installation, the first time you run Smaart, you will be presented with an activation screen requesting an 18-digit license code. To activate Smaart on a computer that is connected to the internet, enter your license code in the fields provided and click the *Next* button. Smaart will prompt for your my.rationalacoustics.com login credentials. Enter the user name and password that you use to log in to the site and then click *Next*. Smaart will automatically fill in three fields for you containing details of the installation that you are registering. The *Name*, *Computer Name*, and *Email Address* can be whatever you want them to be. These are used only to identify the installation on your license management page at my.rationalacoustics.com. When you click the *Activate* button, Smaart will connect to the web site, register the installation, and then activate itself automatically – assuming that you have at least one available installation slot on your license.

If you are activating a Smaart installation on a computer is *not* connected to the internet, you can perform an off-line activation at my.rationalacoustics.com. This requires logging into the web site and entering your computer's unique, 10-digit *Machine ID* (generated by during Smaart installation) to create a unique *Activation Code* for the installation (see *Off-line Activation* on page 195 for details). For more information about Smaart installation, registration and license management general, please refer to *Appendix F: Licensing and Installation* on page 194.

Chapter 1: Fundamental Concepts and Terminology

Depending upon the application, operating Smaart effectively requires a working understanding of a wide range of system measurement concepts and professional audio engineering practices. While it is outside the scope of this document to cover them all, this chapter highlights a few critical concepts that will be of significant assistance in understanding Smaart v8's operation and its application. Readers who wish to deepen their knowledge of these, and other topics related to acoustical measurement and sound system engineering can refer to the reading list in *Appendix A: Applicable Standards and Further Reading* on page 177 for some suggestions on where to go to learn more.

Time and Frequency Domain Analysis

A basic understanding of the relative strengths and differences between time- and frequency-domain analysis is critical to leveraging the measurement power presented in Smaart. The ability to examine a measurement from multiple perspectives is extremely useful in the process of analyzing a signal or system response. Each of Smaart's primary operating modes (real-time and impulse response) includes both time- and frequency-domain measurement and analysis capability.

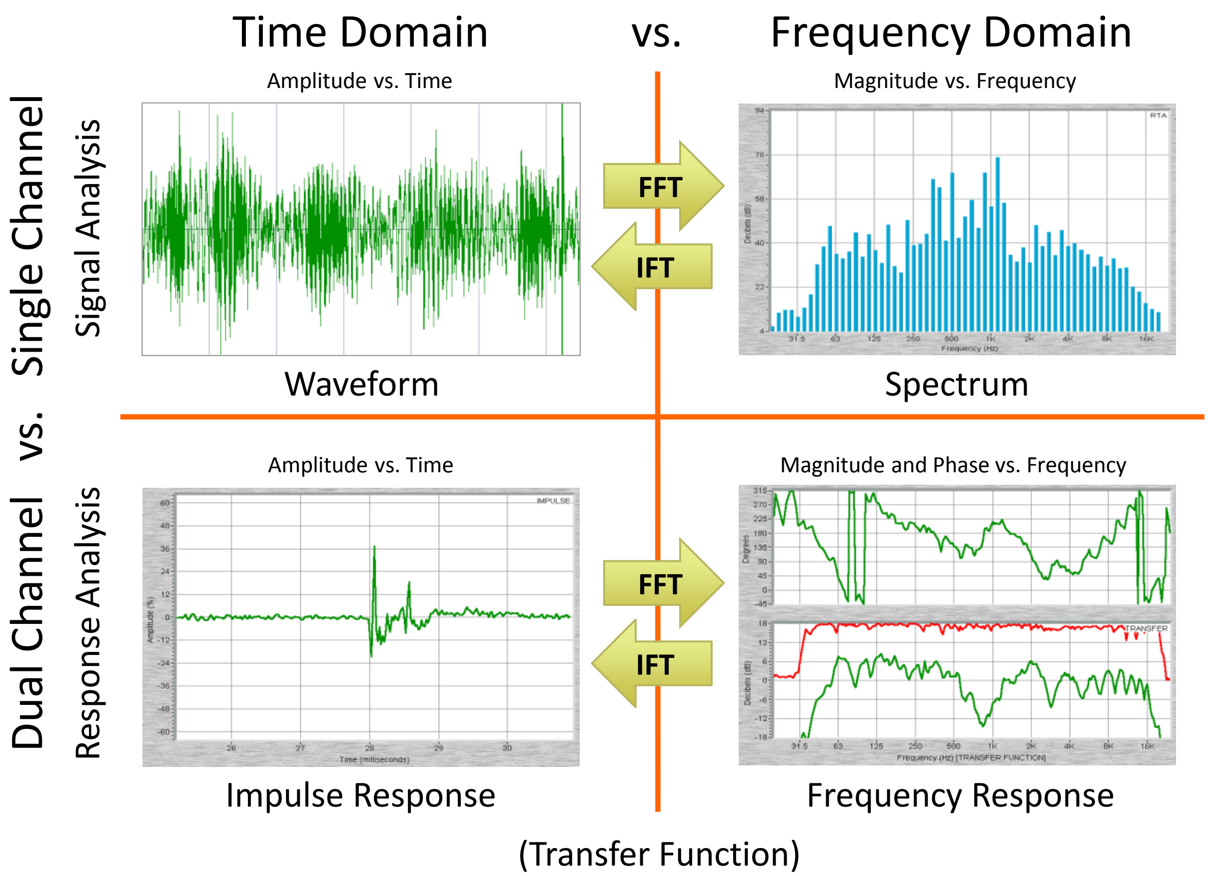


Figure 1: Dual-channel vs. single-channel measurements in the time domain and frequency domain.

The "domain" of a graph or signal refers to the independent variable, usually shown on the horizontal axis of a graph. Audio waveforms, for example, are time-domain signals, where the voltage or digital amplitude of the signal varies over time. Time is the independent variable in this case, so it normally goes on the (horizontal) *x* axis of a waveform graph, with amplitude on the (vertical) *y* axis. On a frequency-domain graph, we normally put frequency on the *x* axis and magnitude on the *y* axis. The exception in both cases is the spectrograph, which has two independent variables, so we orient it whichever direction makes the most sense in a given context.

In recording applications, a time domain graph of an audio signal provides a view of the waveform – a critical view for sound editors. In sound system engineering and room acoustics, a time-domain view of system response (the impulse response) shows the propagation delay through the system and later arriving reflections and reverberation that could potentially be problematic.

Frequency domain analysis of a signal provides a view of its spectrum, which is obviously an extremely useful set of information when analyzing tonal content or looking for feedback. A frequency domain view of system response (the transfer function or frequency response) provides an excellent look at the tonal response of a system as well as its time/phase response by frequency.

Figure 1 provides a very good example of the power of utilizing both time and frequency domain views for examining system response. The frequency response measurement depicts a response with a series of linearly spaced dips and peaks in its magnitude response (lower right). This ripple is a symptom of a problem however, and not the actual problem. The cause of the ripple is clearly identifiable in the time-domain view of the system response as an obvious second arrival in the impulse response, caused by a prominent reflection. Reflections are copies of the direct sound that arrive later in time, after bouncing off of some surface. Mixing two copies of the same signal with a time offset between them results in the comb filter that we can see in the frequency domain view.

Fourier Transforms (DFT/FFT and IFT)

Fourier transforms, named for 19th century French mathematician and physicist Jean-Baptiste Joseph Fourier, are based on the idea that complex signals (such as speech or music) can be constructed from, or broken down into sinewaves of varying amplitude and phase relationships. Fourier transforms are used extensively in audio analysis to find the spectral content of time domain signals. Inverse Fourier transforms (IFTs) reconstruct time-domain signals from spectral data.

There are several different types of Fourier transforms, but the type that we concern ourselves with in Smaart is the discrete Fourier transform (DFT), which works on time domain signals of finite length. The term *fast* Fourier transform (FFT) refers to methods for calculating a DFT more efficiently, most commonly requiring the chunk of signal being analyzed to be a power of two (2^n) samples in length, e.g., 4096 (4K), 8192 (8K), 16384 (16K)… (2^{12}, 2^{13}, 2^{14}…). All FFTs are DFTs, but not all DFTs are *fast*.

Most DFTs in Smaart are power-of-two FFTs (also called radix 2 FFTs or just FFTs). We use arbitrary-length DFTs for some things, notably for impulse response analysis, but since FFTs generally execute much faster, they are very much preferred for real-time operations in particular, or any application where the accompanying restrictions on the precise length of the time record are not a problem.

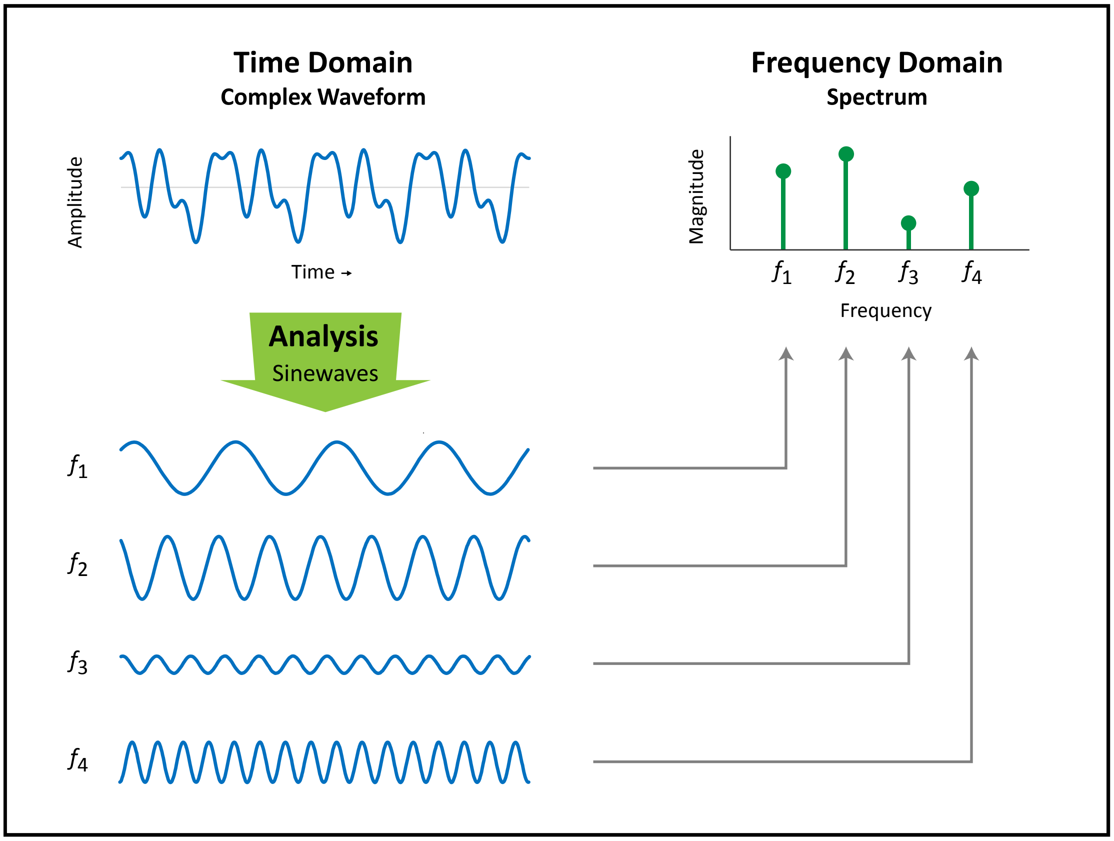


Figure 2: Fourier analysis. The discrete Fourier transform (DFT or FFT) analyzes a complex time-domain signal to find the magnitude and phase of the component sinewaves that make up the complex waveform. The magnitude of each component sinewave can be plotted on a frequency-domain graph to form a picture of the spectral content of the complex signal – the phase data is really only of interest if we have a reference signal to compare it to, or want to synthesize a replica of the original time-domain signal using an inverse Fourier transform (IFT).

Time Resolution versus Frequency Resolution

A key trade-off when working with discrete Fourier transforms (DFT or FFT) is the inverse relationship between time resolution and frequency resolution – as one gets better the other gets worse. Both are a function of the “time constant” (also called the “time window”) of the measurement. The time constant is simply the time that it takes to record enough samples for a DFT of a given size, at a given sampling rate. Longer time windows provide tighter, more detailed frequency resolution (often more than we want at high frequencies) but at the expense of less detailed time resolution.

Time resolution might be the least of your worries if you are doing a long-term average of a signal or a steady-state measurement of a sound system using a statistically random signal such as pink noise. It could however, be an important factor when analyzing a dynamic signal such as speech or music, where you may need to see features of the signal that are very closely spaced in time as separate events. For example, if two drum beats occur within the time constant of a single FFT, the resulting spectrum in the frequency domain includes the energy from both as a single figure at each frequency. If you needed to see each beat as a separate event, you would need to shorten the time window, which would result in more widely spaced frequency bins.

You can calculate the time constant for an FFT (in seconds) by dividing the sampling rate used to record the time-domain signal by the FFT size in samples. For example, the default FFT size for spectrum measurements in Smaart is 16K (16384) samples. A 16K FFT recorded at 48000 samples/second has a time constant of 0.341 seconds (16384/48000) or 341 milliseconds.

$$Time\ Constant = \frac{FFT\ Size}{Sample\ Rate} = \frac{1}{Frequency\ Resolution}$$

Low frequencies have longer cycle times than high frequencies of course – that's what makes them low frequencies – so it makes sense that you have to look at a signal over a longer period of time to resolve them. In fact, the lowest frequency that an FFT (or any other kind of DFT) can clearly "see" is $1/T$, where T is the FFT time constant in seconds. Using the example of a 16K FFT at 48k sample rate, frequency resolution in that case works out to 2.93 Hz (1/0.341).

$$Frequency\ Resolution = \frac{Sample\ Rate}{FFT\ Size} = \frac{1}{Time\ Constant}$$

If you are familiar with the reciprocal relationship between cycle time and frequency in sine waves ($f = 1/t$ and $t = 1/f$), you may have spotted the fact that it echoes the relationship between time constant and frequency resolution in an FFT. In fact, the frequency resolution of an FFT is equal to the frequency of a sinewave that cycles exactly once within the FFT time window. All other frequency bins are at integer multiples (harmonics) of that fundamental frequency, and so knowing the time constant also tells you how far apart the frequency bins are.

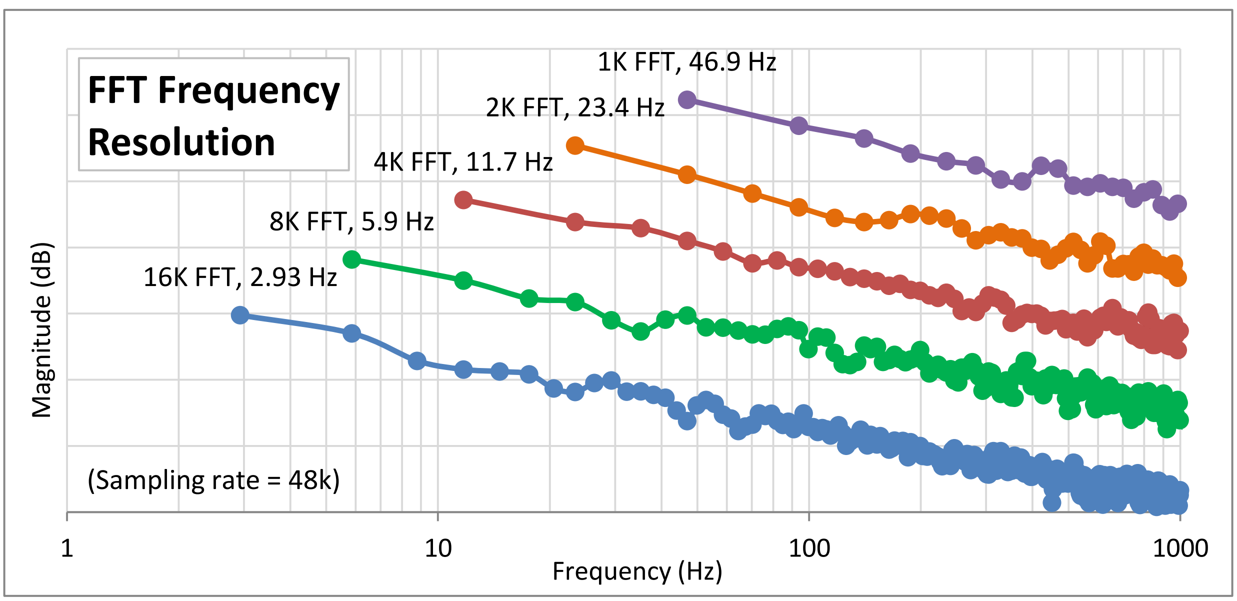


Figure 3: FFT Frequency Resolution shown on a logarithmic frequency scale. Each doubling of FFT size (in samples) doubles the FFT frequency resolution and extends its frequency range an octave lower.

In practical terms, given a sampling rate of 44.1k or 48k, Smaart's 16K default FFT size for spectrum measurements provides very good low-frequency resolution down to the lower reaches of subwoofer frequency ranges, and much greater time resolution than you need for analysis of signals such as pink noise. As regards more dynamic signals such as speech or music, if we recorded 16K FFTs end-to-end for a full minute at 48k sample rate, that works out to just about 176 discrete frames per minute (60 / 0.341 ≈ 176). That tends to meet or exceed the average tempo for most musical genres, meaning that it provides enough time resolution to see the spectral content of individual notes in most cases.

In terms of speech analysis, typical speaking rates for native English speakers range from about 140-180 words per minute or about 200-300 syllables per minute, so a 16K FFT gets you words but not syllables. Dropping the FFT size to 8k would double the time resolution to about a minimum of about 352 frames per second – enough to keep up with insanely fast music or distinguish individual syllables at typical rates of speaking – but does so at the expense of some loss of detail at low frequencies.

A couple of other trade-offs associated with the length of a DFT or FFT are the computational costs, which increase exponentially with size, and the issue of excess frequency resolution at high frequencies when linearly spaced DFT data is plotted on a logarithmic frequency scale. In RTA measurements, the use of fractional octave banding effectively nullifies the excess high-frequency resolution issue and even lower end computers these days can perform real-time analysis using FFT sizes of 16K or even 32K with relative ease.

In transfer function measurements, where computational costs are a bigger problem in general, Smaart's multi-time-window (MTW) feature, attempts to sidestep both problems by using a series of small FFTs at progressively lower sampling rates to deliver approximately 1 Hz resolution at low frequencies without incurring excessively high resolution in the upper octaves. Smoothing the transfer function also helps to clean up excess resolution at high frequencies and works for both MTW and measurements that use just a single FFT size.

Single and Dual-Channel Measurement Techniques

In real-time mode, Smaart performs two basic types of domain measurements: single-channel (signal analysis) and dual-channel (response analysis). Single channel spectrum measurements are signal analysis measurements because all they can tell you is the frequency content and amplitude of a signal. Real-time spectrum analyzer (RTA) and Spectrograph displays are based on single-channel FFT analysis.

Another example would be sound level measurements, i.e., sound pressure level (SPL) or equivalent sound level (Leq). When calibrated to an absolute reference such as SPL, single-channel measurements give you absolute values that are directly comparable to other absolute values and tell you exactly how loud a sound is at a given frequency, or across a given frequency range. They can help to answer questions such as, "How much 1 kHz energy is in that signal," "What is the frequency of that tone," or "What is the SPL at this location in the venue?"

Dual-channel measurements compare two signals to find the similarities and differences between them. Transfer function and impulse response measurements in Smaart are dual-channel measurements that compare the output of a device or system to the input signal that produced it. We can therefore say that we are measuring the response of the system to a given stimulus, and because both signals are known, the spectrum of the input signal becomes almost immaterial. We are also able to precisely measure time relationships between the two signals, enabling us to examine phase relationships and find delay times.

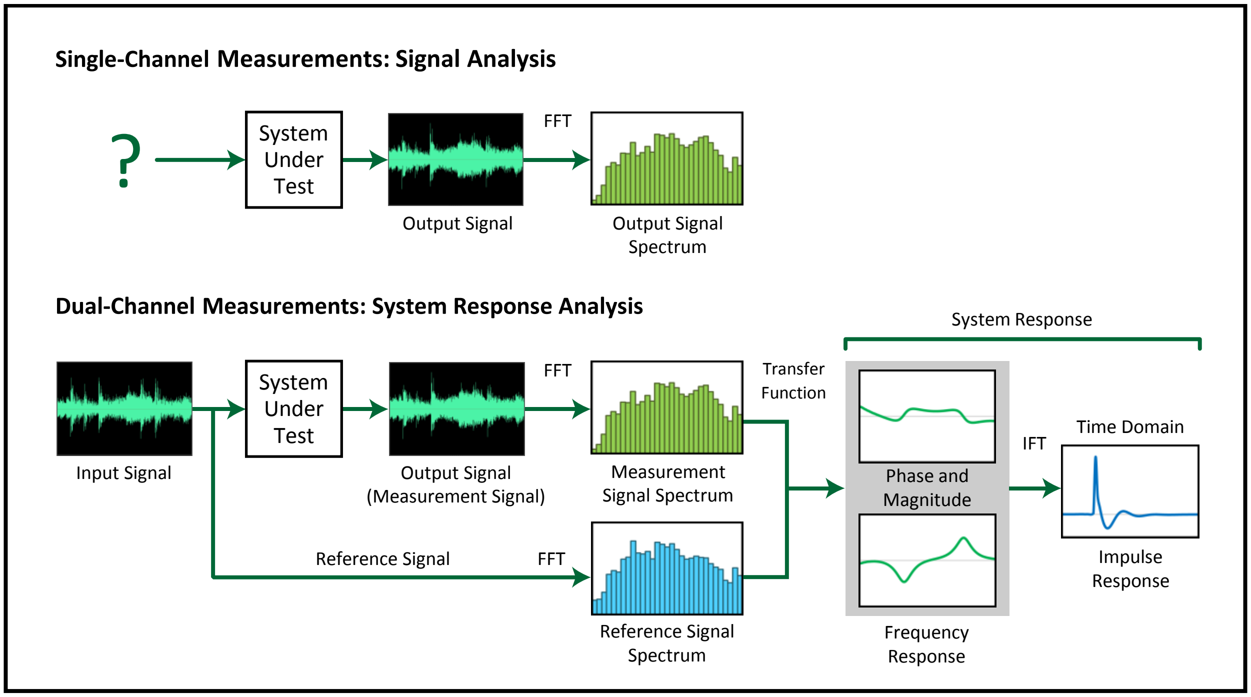


Figure 4: Single-channel vs dual-channel measurements. Single channel spectral measurements analyze the energy content of time domain signals. If the signal being analyzed is the output of a sound system and you happen to know the spectrum of the input signal, you can infer an estimation of the system's magnitude response (only). Dual-channel measurement directly analyzes the input and output signals to provide a more complete picture of system response that includes magnitude and phase response and throughput delay.

Dual-channel methods provide a relative measurement (input vs. output), and can help to answer questions like "What is the crossover frequency in our system," "How much boost or attenuation is there at 1 kHz," or "When is energy from my main speaker system arriving at the measurement mic?"

Both single- and dual-channel measurement can be powerful tools when you understand their individual strengths and weaknesses – what they are measuring, and just as importantly, what they are not. Confusing or conflating the two, however, can lead to poor decisions based on incomplete or incorrect information.

Linear and Logarithmic Scaling

One issue that you run into repeatedly in acoustical analysis is that human perception is logarithmic in nature and covers a relatively huge range of values. Everyone's hearing is a little different but in general, the difference between the threshold of hearing and the threshold of pain – the quietest sounds we can hear and the loudest sounds we can stand – is somewhere around 120 decibels (dB). That works out to six orders of decimal magnitude (the difference between one and one million, e.g.).

In terms of frequency, the audible spectrum for humans is typically defined as 20 Hz to 20 kHz, a range of four logarithmic "decades". Admittedly, many or perhaps most of us are unable to hear across that entire range but it might be safe to say that most people can hear across a range of at least three decades, e.g., from 80 Hz to 8 kHz, which is still a pretty wide range of numbers.

The thing is, we do not hear differences between all those numbers equally in either case. To our senses, the difference between one and two is not the same as the difference between two and three, as it would be if we perceived the world linearly. To us, the difference between one and two sounds (or looks, or feels) more like the difference between two and four, or four and eight, or eight and sixteen… (you get the idea).

Charting audio and acoustic data on logarithmic amplitude (magnitude) or frequency scales does two useful things for us then; it helps to make the wide ranges of values that our hearing encompasses more manageable and it results in a presentation of the data that is often more meaningful in terms of human perception. None of this is to say that linear scales don't have their uses, but for most of the things we do in Smaart, logarithmic scales and units (decades, octaves and decibels), tend to do a better job of showing us what we want to see in a way that makes intuitive sense.

Linear and Logarithmic Frequency Scales

When we talk about Linear and Logarithmic frequency scales (not to be confused with fractional octave banding) we are really just talking about how frequencies are plotted on charts and graphs. On a linear frequency scale, let's say every 100 Hertz (you can pick any number), occupies the same amount of space on the chart as every other. On an octave scale, each octave is the same width as every other, even though the linear frequency range for each band doubles as you ascend in frequency (125 Hz, 250 Hz, 500 Hz, 1 kHz, 2 kHz, 4 kHz…). On a logarithmic decade scale, each power of 10 Hz, (10, 100, 1000, 10,000) is the same width as every other. Logarithmic scales work the same way for any base, but the bases we use for log scales in Smaart are two and ten (octaves and decades).

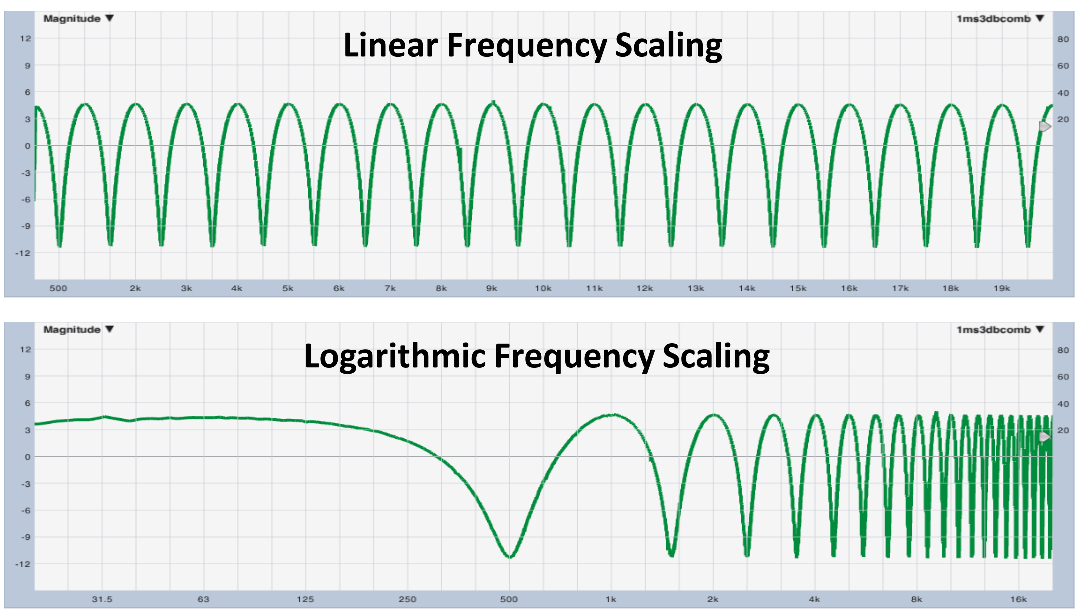


Figure 5: Linear vs Logarithmic frequency scaling. Two views of the same comb filter on a linear and log scaled magnitude graph.

We most often look at frequency on octave or decade scales because these correlates better with our own logarithmic perceptions of sound. However, linear scales are very useful for some things as well, and sometimes correlate better with the underlying physics of sound and acoustics. Charting frequency on a linear scale can make comb filters and harmonic distortion products stand out more clearly since the lobes or peaks are linearly spaced. Another example might be the phase shift associated with a fixed delay, which becomes a straight-line slope on a linear frequency scale.

Note that when you look at FFT data from acoustical measurements or other noisy signals on a log frequency scale, the trace gets fuzzier-looking at higher frequencies. That doesn't necessarily mean there is more noise in the HF. It is a natural consequence of packing more and more linearly-space FFT points into a smaller and smaller amount of chart space. That is one of the reasons for the MTW transfer function option, as noted earlier. Smoothing also helps to reduce visual noise in the HF in transfer function measurements, as does fractional octave banding for spectrum measurements.

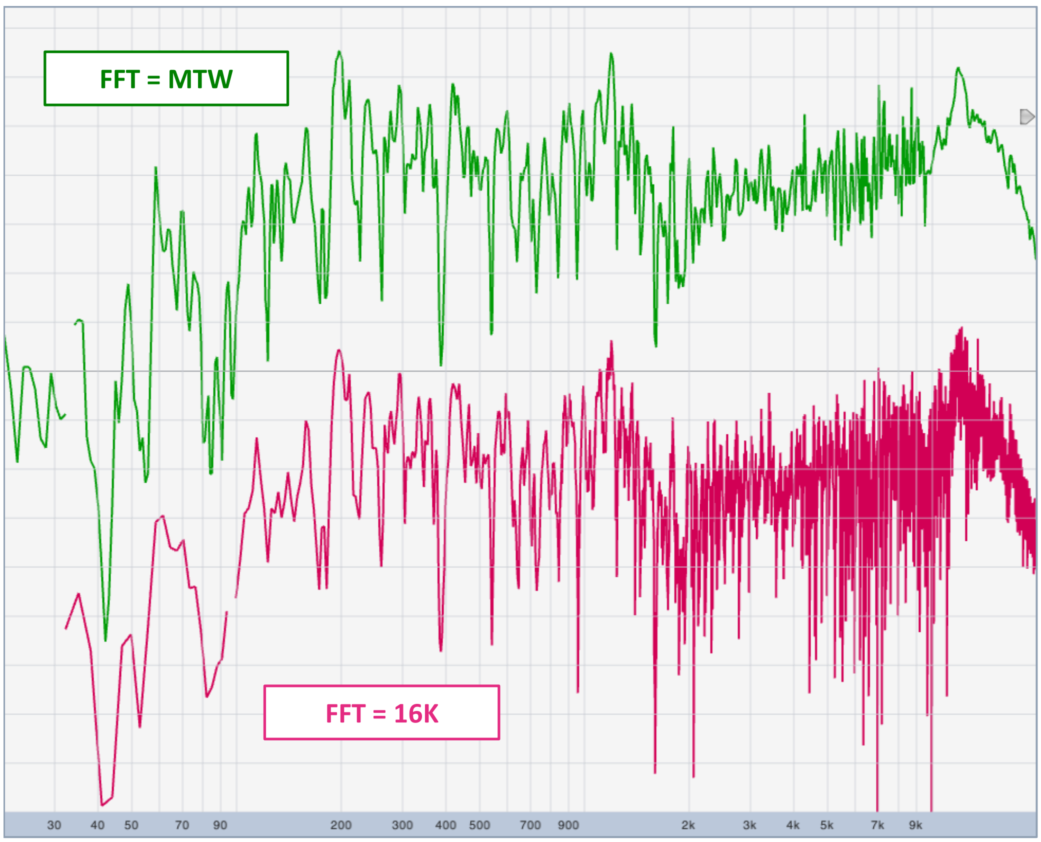


Figure 6: Figure 6: MTW vs 16K FFT transfer function on a logarithmic frequency scale. The MTW uses larger time constants at low frequencies to improve LF resolution while smaller time constants at higher frequencies reduce visual "noise" due to excess resolution.

Linear Amplitude

Linear amplitude, as the name might imply, is amplitude displayed on a linear scale, e.g., volts or digital integer-based amplitude units. In Smaart, the only places that you ever see linear amplitude are linear time-domain charts, where amplitude is displayed as a percentage of normalized full scale. That is to say that the largest positive and negative numbers obtainable from a signed integer of given number of bits (e.g., 16 or 24 bits per sample) are scaled to a range between 1 and -1 (inclusive), with fractional values in between expressed as percentages.

Since you can't take the log of a negative number, the only way to see relative polarity in an impulse response is to use a linear amplitude scale. Also, some people prefer the linear amplitude scale for identifying discrete reflections in an impulse response, and it can be useful for looking at other types of signals as well. A linear amplitude scale tends not be very useful for looking at reverberant decay or for identifying peak structures in the LF range of an impulse response where the length of a waveforms period is spread out of time so much that a clear impulse is not easily discernable.

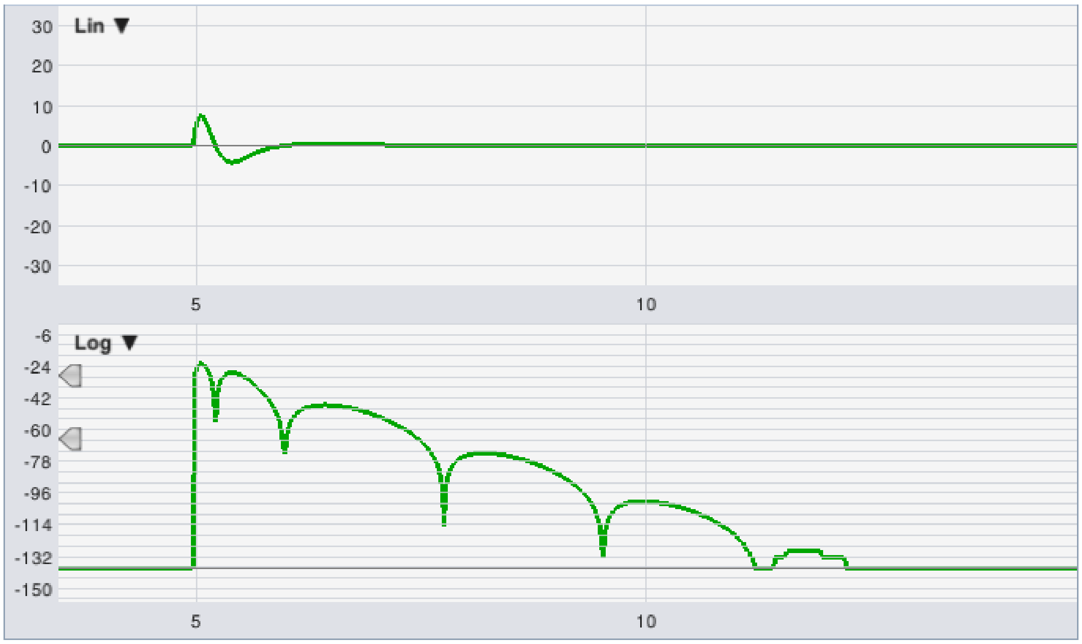


Figure 7: Figure 7: Linear vs Logarithmic amplitude scaling. The impulse response of a bandpass filter is shown on a linear (percentage of normalized full scale) versus logarithmic (decibel) amplitude scale. Notice that only the first two oscillations in the IR are easily discernable on the linear (Lin) view, whereas the Log view clearly shows the first six corresponding lobes.

Decibels (dB)

The decibel is a logarithmic ratio commonly used to express amplitudes, voltages, sound pressure, gain and attenuation and no doubt other things as well. The word literally means one tenth of a Bel. The Bel is named for Alexander Graham Bell, inventor of the telephone (or one of the inventors anyway). Why they called it a "Bel" instead of a "Bell" is a question that someone else would have to answer, but that probably explains why the abbreviation for decibels is written as dB (with a capital B). Although no one seems to use Bels for much of anything – most people would just say 10 dB instead – the formulas for converting to and from decibels may seem less arbitrary if you consider that one Bel represents the logarithm of a power ratio of 10:1 and a decibel is 1/10th of that.

With that thought in mind:

$$dB = 10 \cdot \log_{10}(Power) = 20 \cdot \log_{10}(Amplitude)$$

$$Power = 10^{(dB/10)} = Amplitude^2$$

$$Amplitude = 10^{(dB/20)} = \sqrt{Power}$$

Note that because decibels express a ratio, decibel values must be referenced to something. If no reference is explicitly given, as in the equations on the previous page, the reference is assumed to be one; however it could potentially be any number. To reference dB to a number *other* than one, you simply divide the value that you want to convert to dB by the reference value before taking the log.

In that case:

$$dB = 20 \cdot \log_{10}(v/v_0)$$

where v is some linear value that you want to convert to decibels and v_0 is a reference value.

One common example of this in audio applications is dBu, which references 0 dB to 0.775 Volts. Another might be the AES convention for dB FS (dB Full Scale) which essentially references 0 dB to the square root of 0.5 (0.7071), so that a full-scale, peak-to-peak sinewave has an RMS value of 0 dB instead of −3.01 dB.

Octave and Fractional-Octave Banding (Spectrum Measurements)

Octave and fractional octave banded spectra are another way of reconciling how we hear with what we see on an analyzer screen. On a banded RTA or spectrograph display, each fractional octave band represents the summation of the power at all frequencies that fall within that band. That's why a banded measurement of pink noise looks flat on a banded display, but if you look at the linearly spaced FFT data, you see a signal that rolls off at 3 dB per octave or 10 dB per decade. Each individual FFT bin contains less and less energy as you ascend in frequency but each octave band is comprised of twice as many frequencies, so all the bands add up to an equal number of decibels (given a perfectly pink signal). If you look at a white noise signal, which has equal energy at all frequencies (nominally at least), you would see that it appears flat on an un-banded linear or logarithmic spectrum display, but slopes upward at 3 dB per octave on a banded display.

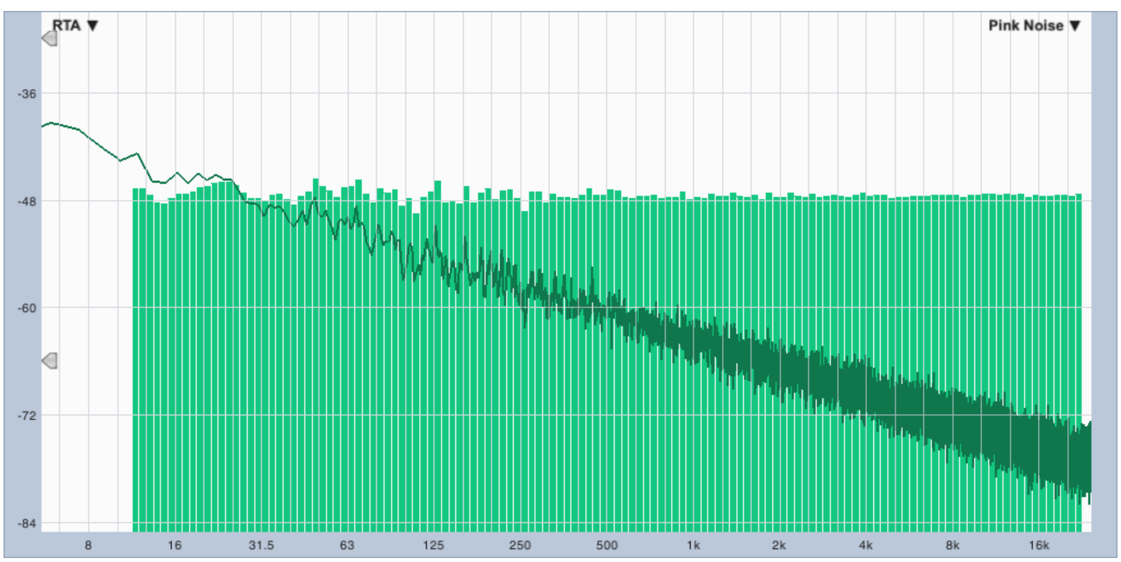


Figure 8: Fractional-octave banding vs FFT data on a logarithmic frequency scale. The lighter green bars show a 1/12-octave banded RTA measurement of pink noise. The darker green line trace is a narrowband (un-banded) view of the same data on a logarithmic frequency scale.

Banded spectrum displays are useful for several reasons. Notably, they can be used in conjunction with pink noise to make a poor man's measurement of the magnitude portion (only) of the frequency response of a device or system. A single channel spectrum measurement can't tell you anything about timing or phase relationships, both of which are important factors in how a system actually sounds, but it could be better than nothing in a pinch, or perhaps as a quick maintenance check of a system that has already been aligned. Another way that banding is useful is just as a way of smoothing spectral data. By summing FFT multiple bins into each band, you immediately start to get a display that is smoother and more stable than watching the individual bins jumping around.

There is a psychoacoustic dimension to banding as well. Pink noise, or 1/f noise as it is called in physics seems to be ubiquitous in nature and in complex systems of all kinds, so perhaps it is not surprising that the long-term average spectra for all kinds of music, across a wide range of genres and cultures, tends to be similar to that of pink noise. Banded spectrum displays may therefore tend to be a natural and intuitive way of looking at the spectral content of music and other signals for that reason.

Smoothing (Transfer Function)

Fractional octave smoothing of transfer function data is useful for comb filter suppression and filtering out noise and other small fluctuations in magnitude and phase response data to help make larger, more audible features and trends in data traces easier to see. Smaart offers two different smoothing functions for transfer function data: fractional octave (logarithmic) smoothing and linear complex smoothing, called FTW (short for frequency-domain time windowing) that is functionally equivalent to windowing the impulse response of a system in the time domain. In general, fractional octave smoothing is the most useful and most commonly used of the two. FTW smoothing is generally reserved for more specialized applications where a windowed impulse response (or rather, its frequency-domain equivalent) is specifically required.

Figure 9: Fractional-octave smoothing for transfer function data.

Fractional Octave Smoothing

Fractional octave smoothing of transfer function data is analogous to fractional octave banding in an RTA measurement. In fact, if you pulled out just the frequencies from a log smoothed trace that correspond to fractional octave band centers and compared them to banded data, you would expect them to be almost identical. The main difference is that in the smoothed data trace, the "bands" overlap, preserving more detail than a bar graph or line trace with data points only at discrete band centers would provide.

In a logarithmically smoothed data trace, each frequency data point is averaged with a varying number of frequency points on either side, depending on band size, frequency, and the frequency resolution of the underlying data. Because the data being smoothed is typically linearly spaced in frequency, the smoothing window widens logarithmically to include more and more adjacent points as frequency increases and the nominal band size grows larger. This helps to tame the HF “fuzziness” inherent in plotting FFT-based measurements of noisy signals on a logarithmic frequency scale, where excessive resolution (relative to how we humans hear) at higher frequencies combined with noise and other environmental factors can make the system response curve difficult to see.

Fractional octave smoothing of magnitude data in Smaart runs directly on magnitude response data, as opposed to complex smoothing, where smoothing is performed on the complex transfer function data *before* magnitude is calculated. This tends to present the magnitude response in a way that correlates better with how we hear than complex-smoothed magnitude, which can suppress reverberant energy that may be audible and emphasize nulls in comb filters that may not be. Phase smoothing in Smaart is always based on complex data, to prevent “wrap” points from being averaged together.

Frequency-Domain “Time-windowing” (FTW), or Linear Complex Smoothing

FTW smoothing is a linear (fixed bandwidth) complex smoothing function applied to the complex transfer function before magnitude and phase data are calculated. In this case, the bandwidth of the smoothing kernel is constant, rather than logarithmically expanding as frequency increases, which means the effects are most noticeable at lower frequencies.

Because the complex data includes both the time (phase) and magnitude response of the system, FTW smoothing affects both the magnitude and phase traces – there are no separate controls for magnitude and phase smoothing as there are for fractional octave smoothing. It is functionally equivalent to applying a data window function to the system impulse response (IR) in the time domain, limiting the effective time constant of the measurement, and then transforming the result back into the frequency domain using a zero-padded FFT.

Impulse response time windows are commonly used in acoustical analysis for such things as excluding problematic reflections from a measurement and/or isolating direct sound from a particular source – e.g., a loudspeaker under test – from later arriving energy emanating from other sources. When a time-windowed IR is transformed into the frequency domain, the practical result is a linear smoothing function, where some later-arriving noise along with comb filters resulting from any reflections that were windowed out are eliminated. FTW smoothing in Smaart emulates the frequency-domain resultant of windowing the IR in the time domain without requiring transforms back and forth between domains.

FTW smoothing in Smaart is applied only to transfer function data measured or captured with *Complex* magnitude averaging. It is considered an advanced feature that must be enabled in *Transfer Function* options to make it available. When FTW is turned on, the global magnitude averaging setting (*Mag Avg Type*) for transfer function measurements is forced to *Complex* and any measurement set locally to *Complex* magnitude averaging will be smoothed as well, along with any captured data traces that were captured with complex magnitude averaging. For more details on transfer function measurement parameters, please see *Transfer Function Measurement Configuration* beginning on page 109.

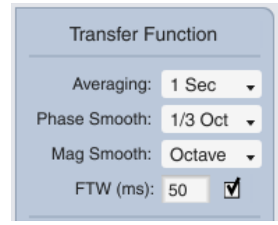


Figure 10: Transfer Function averaging and smoothing controls on the Control Bar in Smaart with FTW shown

FTW smoothing is specified in terms of the equivalent nominal half-window length in the time domain. When FTW is enabled in *Transfer Function* options, a text field for specifying the nominal window size (in milliseconds) and a check-box to turn FTW on and off appear below the fractional octave smoothing controls for transfer function data on the Control Bar in the main Smaart program window(s). Because the equivalent window function in the time domain is fully symmetrical, the maximum half-window size is one half of the FFT time constant. As a general rule however, 25-30% of the FFT time constant might be considered a better practical maximum – e.g., 40-50 ms for an 8K FFT at 48000 samples/sec or 80-100 ms for 16K and so on.

Note that since FTW smoothing limits the effective time window of the transfer function measurement, the effective frequency resolution of the FTW-smoothed measurement depends on the nominal time window, not the time constant of the underlying FFT. For example, a 20 ms FTW window size always has an effective frequency resolution of 100 Hz, regardless of the FFT bin spacing, and data at frequencies below that point should be considered suspect or better yet, ignored. When FTW is turned on, a vertical red line appears on transfer function magnitude and phase graphs indicating this threshold frequency, based on the current nominal time window setting. For more information on the relationship between time and frequency resolution please refer to *Time Resolution versus Frequency Resolution* on page 7.

Averaging

Averaging is used a number of different ways in Smaart, to try and separate useful information from extraneous factors such as noise, reverberation and position-dependent acoustical anomalies. Averaging in Smaart falls into one of two broad categories, temporal or spatial, and there are some different options for each type, depending on the measurement type.

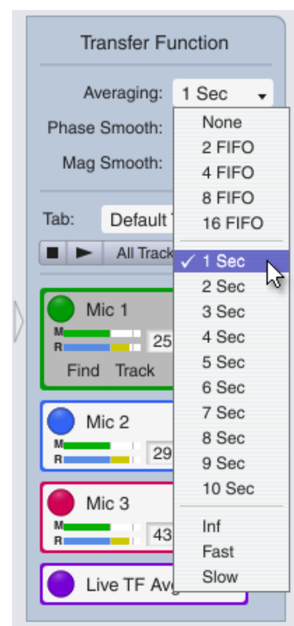


Figure 11: Real-time mode Averaging selector for the active measurement.

Averaging Over Time (Temporal Averaging)

Temporal averaging just means averaging a measurement over some period of time. Typically this is done at a single measurement point or microphone position, although moving-microphone measurements utilizing temporal averaging are sometimes used for specialized applications. In acoustical measurements, a significant amount of noise from various sources gets mixed in with the signal we are trying to measure. The noise components are random, meaning they are different in each individual “frame” of incoming measurement data, and fluctuate quite a bit from one frame to the next. This tends to make the charts jump around a lot and look noisy and hard to read.

Averaging over time increases the signal-to-noise ratio of a measurement through a process known as regression to the mean. The noisy parts of the incoming data, being more random than the signal component, tend to cancel each other out when aggregated over time. The *signal* components, being either stationary features (in the case of steady state system measurements where the signal being measured is not changing rapidly) or at least less random than the noisy parts (when analyzing dynamic signals), tend to average out to themselves, becoming smoother and easier to see.

The trade-off is responsiveness. When analyzing the spectral content of dynamic signals, too much averaging can mask fluctuations that are part of the actual signal and may be things you need to see. In system response measurements, excessive averaging makes the measurement slow to respond to changes in system settings such as equalization and delay adjustments. The trick is to try and use just enough.

For electronic measurements, you can typically get away with very little averaging. In acoustical measurements, the amount of averaging needed varies with background noise levels and user preference. One thing you can do to help speed up the system equalization and alignment process when measuring in a noisy environment is to press the [V] key after making a settings change. This flushes the averaging buffers and restarts the average, so that you don't have to wait for the oldest data to fall out of the measurement before you can begin to see the result of your changes.

Temporal averaging for real-time measurements is set from the Averaging control on the Control Bar that runs down the right side of the main window (see Figure 11). The available options are a mix of types as well as degree of averaging.

- The first four choices in the list are for an equal-weighted simple moving average (called FIFO averaging) of the most recent *2*, *4*, *8* or *16* frames of data. In this type of average, the oldest frame falls completely out of the measurement when a new frame comes in, hence the name "FIFO," for "First In, First Out."
- The options labeled *1-10 Sec* refer to a proprietary averaging method that we call variable averaging, wherein we have tried to combine the most desirable characteristics of FIFO and exponential moving averages.
- *Fast* and *Slow* averaging model the decay characteristics of Fast and Slow exponential time integration used in standard sound level meters. These are first-order exponential averages with time constants of 0.125 and 1.0 seconds respectively.
- Infinite (*Inf*) averaging is a cumulative, equal weighted average with no set period of time. It will simply keep averaging until you stop the measurement or press the [V] key to restart it. You can average over a period of several minutes or even hours if you like, to get the cleanest possible picture of the response of a steady state system or find the long-term average spectrum of a dynamic signal such as speech or music.

Polar vs. Complex Averaging (Transfer Function)

For transfer function measurements, there are two additional options for temporal averaging of magnitude data averaging; *Polar* or *Complex*. Polar averaging might also be called decibel averaging because we first calculate decibel magnitudes for each incoming frame and then take a moving average of the result. Complex averaging keeps two separate running averages of the real and "imaginary" parts of the complex signal and then calculates magnitude and phase from these averages on the back end.

Polar averaging (sometimes called RMS averaging) tends to be the more stable and forgiving of the two, in circumstances where factors as wind, air currents or mechanical movement are present. Complex averaging (also called "vector" averaging) can give you better noise rejection in general and will tend to exclude more reverberant energy than polar averaging.

> Phase averaging in Smaart 8 is always based on a complex data. For temporal averaging in RTA measurements, we always average squared magnitude (power), because we want to see the average power spectrum in that case.

In subjective terms, polar averaging may be the more "musical" of the two options, owing to the fact that it tends to let in more reverberant energy. Complex averaging may tend to correlate a little better with subjective speech intelligibility. This option can be set separately for each transfer function measurement so it is easy to compare them in real time, to see if one gives you a better answer than the other does in a given situation.

Spatial Averaging

Spatial averaging in Smaart works much the same way as temporal averaging. The difference is that in this case, we average measurements taken at different *locations*, rather than measurements made from a single location at different points in time. Spatial averaging can be useful for helping to separate system response from localized acoustical anomalies at a single location or for getting a broader, more statistical picture of background noise or the overall coverage of a loudspeaker than a single position can give you.

If you have multiple microphones and inputs available, you can do spatial averaging in real time. It can also be done by averaging measurement snapshots captured at different locations. We will cover live measurement configuration and working with stored data files in the next chapter. Smaart offers a choice of decibel magnitude (dB) or Power (squared linear magnitude) averaging for both spectrum and transfer measurements however there are some differences in the options for each measurement type.

Power vs Decibel Averaging

Power and decibel (*dB*) averaging refer to what type of data goes into the average. Decibel spatial averaging, sometimes called arithmetic averaging, is a simple average of decibel magnitudes at each frequency. Spatial power averaging is the average of squared linear magnitudes at each frequency with the result converted to decibels. Each has its own strengths and potential weaknesses to keep in mind.

Power averaging would be the typical, and in many cases the required choice for applications such as background noise surveys or evaluating the average power spectrum of sound across a wide area for any other reason. It tends to gives more weight to the loudest sounds and when used for single channel signal analysis, where the focus is more on the sound being analyzed than the response of a system reproducing the sound, it can produce a result that "looks like it sounds." Decibel averaging produces an averaged result where all magnitudes are equally weighted (unless you intentionally give some data more weight). You might say that it tends to give you more of a "consensus" view for all measurement positions than power averaging.

In the context of evaluating sound system frequency response, power averaging works best if all measurements being averaged are approximately equal in level. Its natural bias toward the highest magnitudes means that if one of measurement in average comes in at a significantly higher level than the others, it will tend to dominate the result and could significantly change the shape of the averaged curve. In a decibel average, the higher-level measurement would simply move the entire averaged curve higher on the graph without affecting its overall shape more than any other contributor.

Something to keep in mind about decibel averaging is that it gives as much weight to the nulls in comb filters as it does to the lobes. The nulls, being much deeper than the lobes are tall, can produce dips in the averaged response that *look* like a cause for concern but may be largely inaudible to human listeners at any single location – our ears are generally more sensitive to boosts than cuts and the bandwidth of nulls is much narrower than the lobes, which also tends to make them less audible to humans.

In this case, the natural bias of power averaging toward the highest magnitudes can be helpful as long as the overall levels of all measurements contributing to the average are very similar. This is mainly a concern when averaging spectrum measurement data, where the right answer depends on the purpose of the average and sound level calibration may be a complicating factor in adjusting input levels.

Coherence Weighted and Normalized Power Averaging for Transfer Function Data

With transfer function spatial averages, we begin with a couple of advantages that we don't have for spectrum averages. One is the inherent assumption that the thing being measured is a linear time-invariant (LTI) system – that is, we know we're doing system response analysis in this case and not signal analysis. We also have additional information to work with in, the form of the coherence function, which can be used to weight the data going into the average from individual measurement positions.

Smaart offers three options for transfer function averages: simple unweighted decibel (*dB*) averaging, *Coherence Weighted* dB averaging, or normalized *Power* averaging. Unweighted dB averaging works exactly the same way both transfer function magnitude and spectrum averages. When you select *Power* averaging for transfer function measurements, however, Smaart automatically adjusts the overall level of all individual measurements going into the average according to their average decibel magnitudes in the range of 225 Hz to 8.8 kHz so that they are all approximately equal in level throughout that range.

Coherence weighted dB averaging weights each data point in every measurement in a spatial average according to its coherence value. Coherence is technically an estimation of linearity in transfer function measurements. In practical terms, it tends to be an indicator of a signal-to-noise ratio and so higher coherence suggests that the data is more trustworthy. When averaging data from multiple microphone positions, if one measurement has poor coherence at some frequencies due to nulls in a comb filter, localized reverberant buildup, or perhaps it was taken near the edge of the coverage pattern of a loudspeaker where the HF response was rolling off, coherence weighting should result in the most trustworthy frequencies contributing more to the average than more problematic frequencies.

In practice, normalized power averaging and coherence weighted will tend to produce similar results in otherwise well behaved measurement environments, when comb filtering at individual measurement positions is a significant issue – e.g., due to a floor or ceiling bounce. Coherence weighted dB averaging is the default selection and may be a better choice in cases where background noise and/or reverberation are also significant complicating factors.

Glossary of Terms

Analog to Digital (A/D) Conversion: The process of "digitizing" an analog signal by sampling its amplitude at regular intervals. This process almost always involves limiting the frequency content of the digitized signal to a maximum of one-half the sampling rate, as this provision enables perfect reconstruction of the original band-limited signal from its samples.

Amplitude: In signal processing, the maximum deviation from zero in an alternating signal in either the positive or negative direction, typically expressed in volts for an electrical signal, or as a fractional quantity or percentage of *Full Scale*, in the case of digital signal.

Attenuation: A decrease in the level of a signal. Attenuation can refer to reduction in level for a specified frequency range or a decrease in the overall level.

Block Code: Code obtained after deactivating Smaart 8 from the *About* menu. The Block Code can be used to manually "release" a Machine ID from your license. You will need to know the Block Code if you plan to manually re-register (through my.rationalacoustics.com) a machine ID that has been deactivated.

Coherence Function: In practical terms the coherence function provides an estimation of the signal to noise ratio and the linearity of the system under test in frequency domain transfer function measurements. It is calculated by dividing the averaged cross spectrum of the measurement and reference signals by the power spectrum of the reference signal. The result is a fractional value between zero and one that is typically expressed as a percentage. Larger numbers mean better coherence. Given an ideally linear and noise free system or transmission medium we would expect a coherence value of one (100%) at all frequencies. A value of zero means no detectible correlation between the input and output signals.

Coherence-weighted Averaging uses coherence values to "weight" the contribution from each measurement in a multi-measurement (spatial) average. For example, if one of the measurements contributing to a live trace average has very poor coherence at some frequency, it will have less of an influence on the final averaged trace than measurements with better coherence at that frequency.

Compressor: An electronic device that causes changes in output gain (typically attenuation) as a function of the input level. These devices should NOT be used when making transfer function measurements as they are nonlinear by nature and transfer function measurements assume the system under test is a Linear Time-Invariant system.

Crosstalk: Undesired energy in one signal (or channel) introduced from an adjacent signal or channel.

Data Window Function: A mathematical function that affects the amplitude of a signal over some period of time. Data windows are commonly used to condition a time-domain signal before performing a Discreet Fourier Transform (DFT), to reduce spectral artifacts associated with abrupt truncation of the signal. In theory, data windows can be virtually any shape. In practice, the most useful windows for transforming audio data are smoothly tapering, symmetrical curves such as raised cosine (Hann, Hamming, Blackman) or Gaussian windows that gradually reduce the amplitude of the time domain data at the beginning and end of a finite time/amplitude series to zero or nearly so.

Decay Rate: The rate at which a system decays from an excited state after cessation of a stimulus signal. In acoustics, this quantity is usually evaluated on the basis of specified frequency ranges and expressed in either decibels per second, or as the amount of time it would be required for the signal to decay 60 decibels at the observed rate of decay. (see Reverberation Time)

Decay Time: See Reverberation Time.

Decibel: The decibel, often abbreviated as dB, is a logarithmic ratio between two values. In electronics and acoustics, decibels most commonly refer to the ratio between a given amplitude value and the number 1, where some reference value such as the maximum output of an A/D converter (dB Full Scale or dBFS) or the threshold of audibility for human hearing (for dB SPL) is scaled to equal 1. The decibel value for an amplitude is then calculated as: $\text{dB} = 20 \cdot \text{Log}_{10}(A) = 10 \cdot \text{Log}_{10}(A^2)$, where A is linear amplitude. In this case, amplitude values greater than one yield positive decibel values and numbers smaller than one become negative dB values. This is why dB FS values are most often negative and dB values in sound level measurements are nearly always positive.

Digital Full Scale: See *Full Scale (FS)*.

Discrete Fourier Transform (DFT): A mathematical technique for determining the spectral content of complex waveforms. The DFT essentially compares the signal being analyzed to a series of sine and cosine waves at regularly spaced (harmonic) intervals to determine how much energy is present at each harmonic frequency. The spacing between frequencies or *frequency resolution* of the DFT is a function of its size in samples and the sampling rate used to record the signal being analyzed. Plotting the amplitudes of the energy found at each frequency an *x*/*y* graph produces a visual representation of the spectral content of the original time-domain signal.

Domain: In signal processing, the term “domain” refers to the independent variable of a signal. By convention, when graphing a signal the independent variable is typically placed on the horizontal (*x*) axis of the plot with the dependent variable on the vertical (*y*) axis. For example in Smaart, an impulse response display, with time (the independent variable) on the *x* axis and amplitude (the dependent variable) on the *y*-axis, is referred to as a *time domain* display. Similarly, Spectrum and Transfer Function displays where magnitude or phase shift are plotted as a function of frequency are called *frequency-domain* displays.

Dynamic Range: The difference in level between the highest and lowest signal a system can accept or reproduce, for example the range between the noise floor and the clipping voltage of an amplifier, typically expressed in decibels.

Equalizer (EQ): A device with some number of filters used to change the relative gain or attenuation of a signal at some frequencies but not others. The term “equalizer” comes from the fact that a primary application for this type of device is to “flatten out” (i.e. equalize) the most offending lumps and bumps in the frequency response curve of a sound system to make it more acoustically transparent. Equalizer filters may be “active,” providing either boost or attenuation in the filter’s passband, or “passive” (attenuation-only). The gain of each filter is usually independently adjustable. The center frequencies and bandwidths of filters can be variable or fixed. A filter bank made up of bandpass filters with fixed frequencies and bandwidths, e.g., on 1/3-octave intervals is commonly referred to as a *graphic* EQ. When the frequencies and bandwidths for each filter in a filter bank are variable along with the gains, it is called a parametric EQ.

Fast Fourier Transform (FFT): A *Fast* Fourier Transform is a special case of a *Discrete* Fourier Transform (DFT) that is optimized for ease of computation. In practice this typically involves specifying the lengths of time domain signals to be a power of 2 samples in length (e.g., 16, 32, 64, 128, 256...). This limitation allows some shortcuts to be used in calculating a DFT on a digital computer using binary math that significantly reduce the number of computational operations required, resulting in much faster execution times.

FFT/DFT Frequency Resolution: The frequency resolution of a Discrete Fourier Transform (DFT or FFT) is determined by dividing the sampling rate used to record the time-domain data by the number of samples in the time record being transformed. For example a 32K (32768-sample) FFT of a time record sampled at 48000 Hz, will have a frequency resolution of 1.46 Hz, meaning there will be one linearly spaced frequency data point every 1.46 Hz along the frequency axis.

FFT/DFT Time Constant: The amount of time it takes to collect all the samples required for a single FFT frame of a given size at a given sampling rate. The time constant of an FFT, also called the time window, can be calculated by dividing the FFT size by the sampling rate. For example, a 32K (32768-sample) FFT sampled at 48k samples/second has a time constant of 0.683 seconds.

Full Scale (FS): The term full scale has two possible meanings in digital signal processing. Normalized full scale refers to the maximum amplitude of a digital signal sampled at a given integer sample word size (bits per sample), scaled to +/- 1.0, such that 0 dB corresponds to the maximum possible <u>peak</u> signal value and all lesser decibel values are negative. A second convention preferred by the Audio Engineering Society (AES) references 0 dB to the <u>RMS</u> value of a full-scale peak-to-peak sinewave (i.e., 0.7071 rather than 1.0). In Smaart 8, all FS decibel values are referenced to *normalized* full scale, meaning that the RMS power of a full scale sinewave is -3.01 dB. When using a linear amplitude scale, Full Scale amplitude values are typically given as percentages, where 100% = 1.

Graphic Equalizer: An equalizer with some number of bandpass filters used to change the gain or attenuation of a signal at pre-selected frequencies. The bandwidths and center frequencies of the filters are typically spaced on octave or fractional octave intervals and usually are not adjustable by the end user. The term “graphic” comes from the fact that a series of linear faders arranged side-by-side are typically used to adjust the gains of individual filters so that the knobs on the faders forms a sort of a rough graph suggestive of the unit’s response curve. In practice however, interactions between adjacent filters can often make the term something of a misnomer.

Impulse Response: The signal that describes the response of a system to an ideal impulsive stimulus in the time domain. The impulse response of linear time invariant (LTI) system is the inverse Fourier transform of its frequency-domain transfer function.

Latency: In signal processing, latency refers to the throughput delay for a device or signal chain. All digital signal processing devices introduce some amount of latency into a signal chain.

Linear Scale: A set of values in which values are evenly spaced. On a linear scale, each value (or unit) has equal dimension and each integer multiple of a base number or unit represents an equal stride, so that 1, 2 and 3 are all equal steps, as are 10, 20, 30...

Linear Time Invariant (LTI) System: It is not uncommon for descriptions of linear time invariant systems to run pages in length and include a lot of scary math. But in simple terms, LTI essentially means that a given input will always produce a predictably and proportionally scaled output and should always require the same amount of time to work its way through the system. For example, if you put in a five and get out a 10, then putting in 10 should get you 20, and throughput delay (latency) should be the same in both cases. Gain and latency through the system need not be the same for all frequencies, but they should be consistent for any given frequency. Most of the components in a sound system, with the exception of intentionally nonlinear processors such as compressors, limiters and special effects, are intended to be LTI systems. From our point of view, a really useful property of LTI systems is that they can be completely characterized by their transfer function in the frequency domain and/or their impulse response in the time domain.

Logarithmic Scale: A scale on which each power of a given number (e.g., ten) is given equal dimension. On a logarithmic scale, orders of magnitude, e.g., 10, 100, 1000, 10,000... (a.k.a., 10^1, 10^2, 10^3, 10^4...), are equal intervals. On a base 10 logarithmic scale, orders of magnitude are often referred to as "decades." On a base 2 scale, each stride is essentially one octave.

Machine ID: A unique code assigned to each installation of Smaart 8 during the installation process.

Magnitude: 1. A number assigned to a quantity so that it may be compared with other quantities. 2. The absolute value of amplitude. As a convention, we most often use the terms *amplitude* to refer to linearly scaled quantities and *magnitude* when discussing amplitudes cast in logarithmic units such as decibels or orders of magnitude.

Nyquist Frequency: Named for Harry Nyquist, a pioneer in the field of digital signal processing (although it wasn't called that at the time), the Nyquist frequency is a relative quantity equal to one half of the sampling rate used to record a digitized signal. The Nyquist frequency is important because it represents the theoretical limit for the highest frequency that can be accurately reconstructed from a sampled signal. (In practice, the real-world limit tends to a little lower due to the difficulties associated with creating a perfect brick-wall low pass filter for anti-aliasing and signal reconstruction.)

Octave-Band Resolution: On an octave or fractional octave band display the aggregate power for all the frequencies within each band is summed and displayed as a single value per band. It is a common practice to display octave-banded data as a bar chart, rather than a line trace, to better convey the idea that each value shown on the graph represents the total power across a *range* of frequencies, not just a single frequency point at the band center. Note that by convention, the *nominal* center frequencies given for ISO standard octave and 1/3-octave bands are slightly different than the *exact* band center frequencies in most cases, but they're close.

Overlap: For Smaart's purposes, the term *Overlap* refers to the amount of data each successive FFT Frame shares in common with the one before. Overlapping FFT frames are analogous to shingles on a roof. When no overlap is used, each new FFT frame begins where the last one stopped, as beads on a string.

Parametric Equalizer: An equalizer or digital filter bank in which the relative gain or attenuation, as well as frequency and bandwidth of individual filters are independently adjustable.

Phase Shift: A timing difference in a signal (relative to some reference) at one or more frequencies, typically expressed in degrees, where 360° = one full cycle at a given frequency.

Pink Noise: A random (or pseudorandom) signal in which, over a given averaging period, each Octave-band (or other logarithmically spaced interval) contains an equal amount of energy.

Propagation Delay: The time it takes for sound to travel from one place (such as a loudspeaker) to another place (e.g., a measurement microphone).

Reverberation Time: In acoustics, the amount of time required for audio energy introduced into a system (typically a room) to diminish, or decay by 60 decibels following the cessation of a stimulus signal used to excite the system — e.g., a balloon pop, gun shot or terminated pink noise. It is normally stated band-by-band for individual octave bands. By convention, decay times are normalized to the time required for 60 dB of decay at an observed rate of decay, regardless of the amplitude range actually measured. 60 dB decay time is often referred to as “RT60” or “T60", which is sometimes a source of confusion. And just to confuse things a little more ISO 3382 specifies that it should be called T20 or T30, where the “20" and “30” refer to the decay range actually measured. The main thing to remember is that all of the above refer to 60 dB decay time within a stated frequency range. When a single-number reverberation time is given, according to the ISO standard it should be the average of the reverberation times for the 500 Hz and 1 kHz octave bands, also called “T_{mid}."

RT60: See Reverberation Time.

Sampling Rate: The number of times that the amplitude of a signal is measured within a given period of time in analog-to-digital conversion. For audio-frequency signals, sampling rate is typically expressed in samples/second or Hertz.

Signal: Strictly speaking a signal can be any set of values that depends on some other set of values. In signal processing, the independent variable, e.g., time or frequency, is said to be the *domain* of the signal. In audio and acoustics the things we most commonly think of as signals are time domain signals, where voltages or numeric values representing amplitude (the dependent variable) vary over time (the independent variable). But by strict definition, most of the things we see in Smaart could also be called signals, including transfer function Magnitude and Phase and RTA displays where relative energy or phase shift are presented as a function of *frequency*, rather than time.

Sound Pressure Level (SPL): The RMS level of pressure waves in air expressed in decibels, referenced to the approximate threshold of audibility for human hearing where 0 dB is approximately the quietest sound the average human being can detect (and 1 Pascal H 94 dB). SPL is typically integrated over some period of time, e.g. using the Fast and Slow time integration settings found on all standard sound level meters. It is most often weighted by frequency to reflect perceptual characteristics of human hearing using standard A or C weighting curves specified in IEC and ANSI standards related to the measurement of sound levels.

Speech Transmission Index (STI): An objective estimate of the intelligibility of human speech transmitted through a transmission medium or system under test. It is based on the relative loss of very low frequency modulation in higher frequency carrier waves. STI is stated as a fractional value between from 0 to 1, where 0 represents a completely unintelligible result and 1 denotes excellent intelligibility with no loss of modulation in transmission. STI is calculated from the modulation transfer functions (MTF) for 7 octave bands from 125 Hz to 8 kHz, and evaluates 13 modulation frequencies in each band. The results for all modulation frequencies in each band are combined into a band-by-band Modulation Transfer Index (MTI) then weighted and summed to produce a single-number STI figure. Separate estimates for the male and female speech may be obtained using different weighting tables when calculating STI. Unless otherwise stated, weighting for male speakers is presumed.

Spectrograph: A three-dimensional data plot, displayed in two dimensions with color representing the third dimension (or z-axis). The spectrograph is a topographical representation of the once-common waterfall display.

Spectrum: The frequency content of a given signal.

Speed of Sound: The speed at which sound waves propagate through a transmission medium such as air or water. This quantity has dependent actors such as temperature and density of the material of propagation. Useful rule-of thumb values for the speed of sound in air at room temperature are 1130 ft/sec, or 344 m/sec. In Smaart the speed of sound is used primarily to calculate distance equivalents for time differences.

STIPA (STI for public address systems) is a less rigorous variant of STI (see above) intended specifically for use in measuring public address systems. STIPA has been validated only for male speech. The only functional difference in how STI and STIPA are measured is that instead of evaluating 13 modulation frequencies in each octave band, STIPA uses only two frequencies per band to cut down the time required for direct measurement of the modulation transfer functions (MTF). Since Smaart calculates the MTF of a system indirectly from its impulse response and all 13 modulation frequencies required for STI are also used for STIPA (just not in every band) there is no particular advantage to using STIPA in Smaart.

T60: See Reverberation Time.

Transfer Function: The frequency (magnitude and phase) response of a system, function or network. The transfer function of the linear time invariant (LTI) system can be measured directly, using techniques such as dual-FFT transfer function measurements that compare the output of the system to its input signal in the frequency domain, or by taking the Fourier transform of the system’s impulse response.

Time Constant: In physics and engineering the term *time constant* is most commonly used to denote a time span between reference or threshold points in continuous processes, such as rise or decay time in the step response of filters, heating and cooling times in thermal systems and lag times in mechanical systems. In the context of acoustic measurement we typically use (or perhaps misuse) the term to mean the total time required for discrete processes, such as the time it takes to collect enough samples for an FFT and/or the time span of an impulse response measurement. This is to say that we tend to use the term interchangeably with *Time Window*.

Time Window: The amount of time required for and/or represented by a measurement or other process. Often used interchangeably with Time Constant (see above).

White Noise: A random (or pseudorandom) signal in which, over a given averaging period, each frequency has equal energy. White noise is a common test signal in electronics. It is seldom used in testing systems that include loudspeakers because it has so much high-frequency energy that it can easily damage HF components of the system, and human hearing as well.

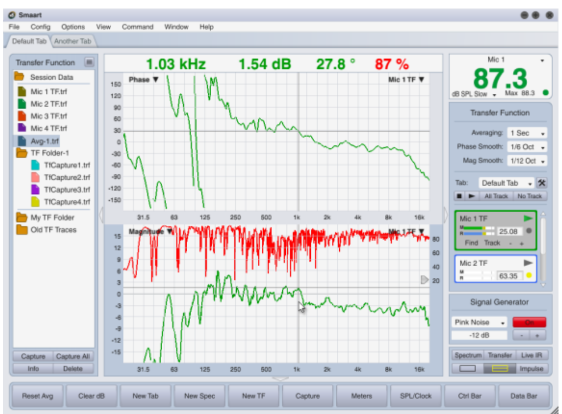
Chapter 2: Finding Your Way Around in Smaart

Multiple Windows and Tabs

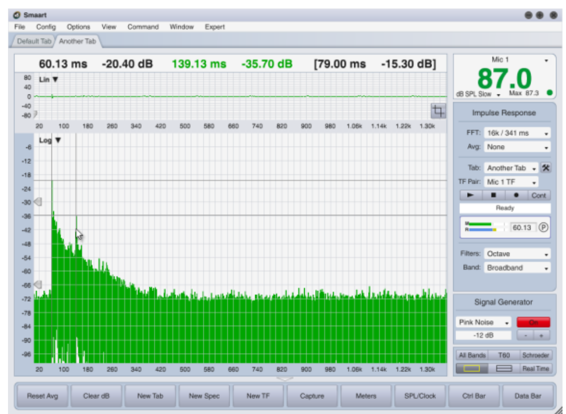
Smaart v8 can run in multiple windows and each window can host multiple "tabs." If you are familiar with Smaart v7, each tab in version 8 is almost like a complete copy of Smaart v7, with its own live measurements, graphs and window layout. You can switch between tabs using the tab-shaped buttons in the Tab Bar at the top of the window, just below the menu bar.

Two Distinct Measurement and Analysis Modes

Smaart operates in two distinct measurement and analysis modes: *Real-Time* and *Impulse Response*. You can toggle between measurement modes using the mode buttons in the *Control Bar*, via the *Real-Time Mode* and *Impulse Response Mode* commands in the *View* menu, or by using the [I] and [R] hot keys. Yet another way to get back and forth is by recalling a view preset based on one mode or the other, but perhaps we are getting ahead of ourselves.



Real-Time Mode



Impulse Response Mode

Figure 12: Two distinct operating modes, real-time and impulse response.

In both modes, you can actively measure and display both frequency- and time-domain data. The fundamental distinction between them is their operational focus. Real-Time mode is an environment for measuring and capturing spectrum and transfer function measurements – often in multiples – in real time, and is optimized for on-site system alignment and mix engineering work. Impulse Response mode provides a robust set of tools for measuring and analyzing the acoustical properties of systems and rooms, including analysis of reverberation times, early to late energy ratios, and speech intelligibility metrics. We will explore the user interface for each mode in detail in later chapters. For now, we will focus on things they have in common.

Common User Interface Elements

At first glance, the default window layouts for real-time and impulse modes look very similar. Each has a Control Bar on the right side of the main window with a large numeric signal level/sound level meter at the top. To the left of the Control Bar is the main graph area with the cursor tracking readout above it, which takes up the bulk of the window area. At the bottom of the window is a row of user-configurable buttons that we call the Command Bar. Real-Time mode tabs also have a Data Bar area on the left. Each of these components can be hidden by clicking the triangular buttons in the border areas that separate them from the graph area or by using menu commands or keyboard shortcuts listed in the *View* menu. These buttons remain visible in the window border when the corresponding area is hidden. Clicking the button for a hidden area again will restore it to the tab. The tab bar at the top of the window enables you to switch between the tabbed “pages” in windows with multiple tabs.

The Tab Bar

Smaart can run multiple windows and each window can host multiple tabbed workspaces that we refer to simply as tabs. Each tab includes its own measurements, screen layout, and graph assignments. You can navigate between them by clicking the tab-shaped buttons that normally appear in the upper portion of Smaart main program windows just below the menu bar. You can move a tab from one Smaart window to another by clicking on its button in the Tab Bar with your mouse and dragging it to another window, then releasing the mouse button to drop it.

If you are not using multiple tabs or are not switching between tabs very often, you can hide the Tab Bar to make more room for graphs by selecting *Tab Bar* from the *View* menu or pressing the [A] key on your keyboard. Repeating either of these actions will restore the Tab Bar when it is hidden. Note that when the Tab Bar is hidden, you can still switch between tabs using the *Tab* selector on the Control Bar.

Graph Area Allocation

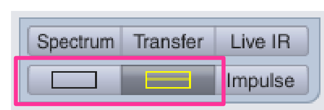


Figure 13: Graph Area allocation buttons

The main graph area in both real-time and impulse modes can be subdivided into multiple panes. In real-time mode, it can be divided into two panes at any time, each of which can be assigned any of four real-time frequency-domain graph types: *RTA*, *Spectrograph*, *Magnitude* or *Phase*. When a transfer function *Magnitude* or *Phase* graph is visible, you also have the option of adding a third pane with a live impulse response display by clicking the *Live IR* button.

In impulse response mode there is a small graph pane at the top of the main graph area, just below the cursor readout, and is always visible. This graph is used for navigation purposes. The lower portion of the graph area can again be allocated as one or two graph panes, each of which can be assigned any of the six main graph types for IR mode (*Lin*, *Log*, or *ETC* time domain displays, *Frequency*, *Spectrograph*, or *Histogram*). The *Live IR* pane in real-time and the navigation pane in impulse mode are restricted to time-domain graph types only (*Lin, Log* or *ETC*).

Active Graph Pane

When multiple graphs panes are present on a tab, one of them is always considered to be the *active* graph. Clicking on a graph pane with your mouse selects it as the active pane and you can tell which pane is active by the color of the margins around the graph.

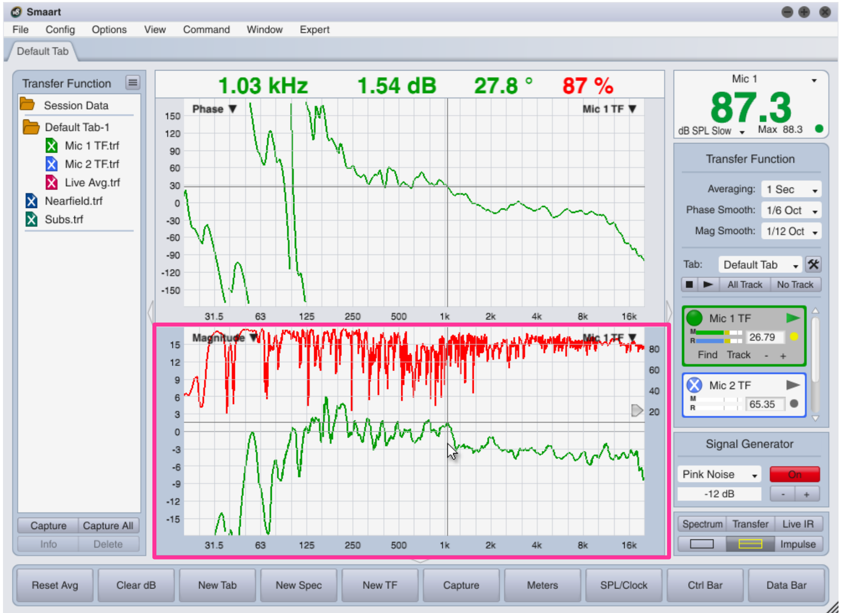


Figure 14: Active graph pane selection

In real-time mode, the active graph pane selection determines which controls you see in the Control Bar and which set of captured data traces you see in the Data Bar on left the side of the window (Spectrum or Transfer Function). The default color schemes set the highlight color for the active graph margins to match the background color of the Control Bar and Data Bar, to emphasize the fact that these three things go together. The active graph pane is the source for data capture operations in Real-Time mode and it is the target for most menu or keyboard commands that affect what you see on a graph, such as zoom keys or cycle *z*-order commands.

Graph Type Selection

The graph type currently assigned to each graph pane in Smaart is shown in the upper left corner of the pane. Clicking on this label pops up a menu showing you the available graph types for the selected graph pane. You can change the graph type assignment by selecting another graph type from the menu.

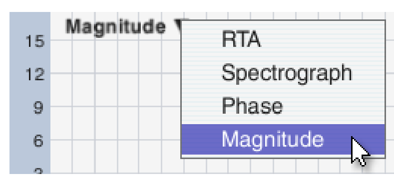


Figure 15: Graph type selection.

Secondary Graph Controls

In addition to the graph type selector in the upper left corner of each pane, most graphs in Smaart have additional controls in the upper right corner. Additionally some graphs include movable widgets that control thresholds for spectrograph dynamic range and coherence blanking, as applicable.

In impulse response mode, the frequency-domain graph has a menu in the upper right corner to select *Smoothing* for the frequency response trace. The IR mode spectrograph has on-graph controls for *FFT* size and *Overlap* in the upper right corner along with a *Calc* button that recalculates the spectrograph. The navigation pane also features a movable widget at the bottom of the graph that lets you move the time-zero point on an impulse response measurement for display purposes.

In Real-Time mode, the name of the top trace (the one that is currently displayed in front of all the others, at the top of the *z*-axis stack) on each graph appears in the upper-right corner of its graph pane. You can cycle the z-axis stacking order forward or backward by pressing the [Z] key or [Shift]+[Z] on your keyboard. On all real-time graphs except the Live IR graph, clicking on the top trace label opens the legend box for the graph (see *Graph Legends* on page 71 for details).

Threshold Widgets

The spectrograph displays in both real-time and impulse response modes feature a pair of arrowhead-shaped widgets positioned on the left edge of the graph. You can click and drag this with your mouse to set the minimum and maximum values for the spectrograph's dynamic range. The spectrograph range control widgets are echoed on the RTA graph in real-time mode and on Log IR and ETC graphs in impulse response mode. On the transfer function magnitude graph in real-time mode, a threshold control widget on the right edge of the graph sets the coherence-blanking threshold.

Zooming

You can zoom in and out on any of the graphs in Smaart with your mouse, using hot keys or by means of user-definable zoom range presets. Press the plus and minus keys ([+] and [-]) to zoom in and out on the (vertical) *y* axis of any active graph in Smaart except the navigation pane in impulse response mode, which has a fixed *y* range. Holding down the [Alt/Option] key while pressing plus or minus will zoom in/out on the (horizontal) *x* axis of the active graph. Holding down the [Ctrl/Cmd] key while pressing plus or minus will zoom in or out on the *both* the *x* and *y* axes of the active graph.

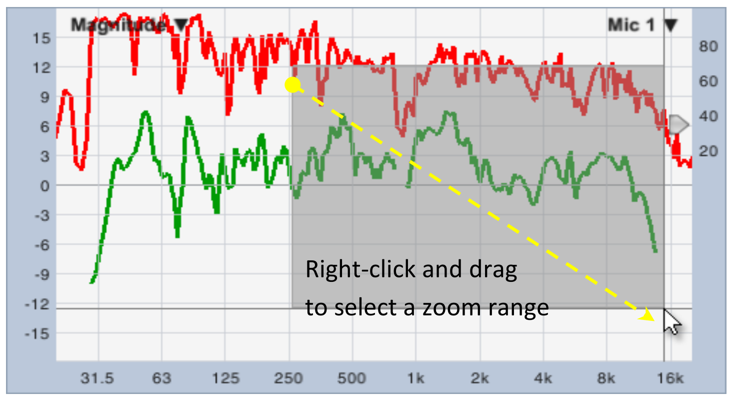


Figure 16: Mouse zooming (aka rubber band zooming). Right-click ([Ctrl]+click on Mac) and drag with your mouse on any graph to select a zoom range for display.

You can also right-click ([Ctrl]+click on Mac) and drag with your mouse on any graph to draw a "rubber-band box" around the area you want to zoom in on. When you release the mouse button the selected area will expand to fill the entire graph. Clicking in the margin of any plot that you have zoomed in on restores it to its default zoom range.

Zoom Presets

Zoom presets enable you to set the *x*-axis range of a graph (the time or frequency axis) to a predefined range. Zoom presets are activated by selecting *Zoom* > *Zoom 1-4* from the *Command* menu or pressing [Alt/Option] + [1-4] on your keyboard. Selecting *Default* from the *Zoom* menu in the main Smaart window, pressing [Alt/Option] + [5] or simply clicking in the margin of a plot restores it to its default *x* and *y* ranges.

Zoom preset ranges are set from the *Zoom* tab of the *Options* dialog, accessible by selecting *Zoom* from the *Options* menu. There are three groups of settings for time- and frequency-domain graphs.

- Settings in the *Frequency* section of *Zoom* options define preset zoom ranges for frequency-domain graphs in both real-time and impulse response modes.
- Zoom range settings in the *Absolute Time* section apply to time-domain graphs in impulse response mode only.
- *Relative Time* settings define +/- time ranges for the transfer function *Live IR* display (centered on the delay time setting for the active transfer function measurement).

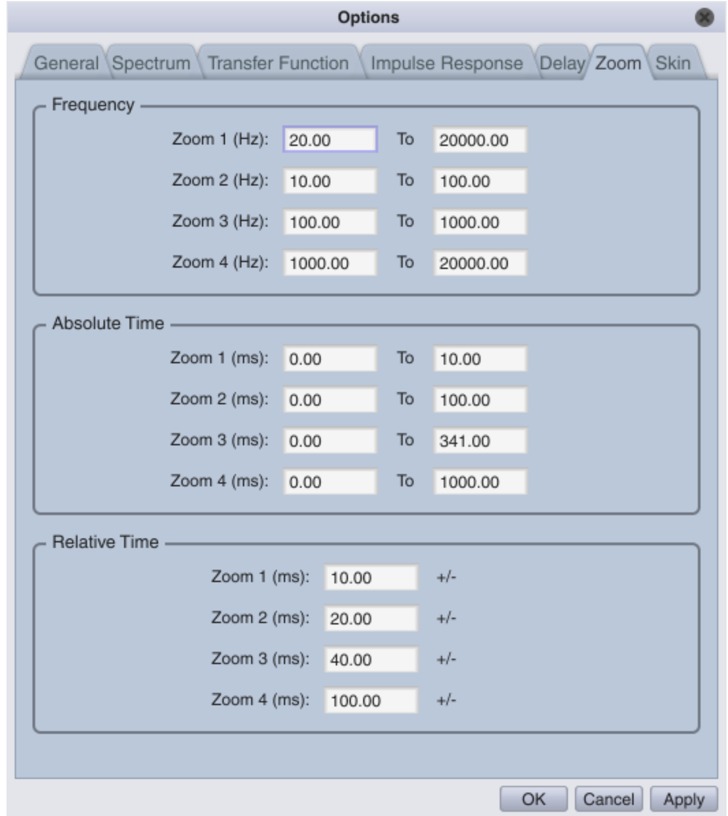


Figure 17: Zoom preset options

Cursor Readout

One of the most important analysis tools in Smaart is the humble mouse (or other pointing device). In addition to clicking buttons, interacting with control widgets and making menu selections, the mouse cursor is used to find the precise frequency, amplitude, phase or time coordinates (as applicable) of any point that interests you on any real-time or impulse response mode chart. When you position the cursor over any data plot in the graph area, the cursor readout above the main graph area displays the coordinates of the mouse cursor in amplitude, frequency or time units, depending on the chart type.

The cursor readout can also display *Wavelength* or *Note ID* on frequency-domain charts and distance units on time-domain graphs. These options are located on the *General* tab of the *Options* dialog window, accessible by selecting *General* from the *Options* menu.

Mouse Cursor Tracks Data (or Not)

As you move your mouse cursor over any graph in Smaart – assuming there is visible data on the graph – you will notice horizontal and vertical lines extending all the way across the plot, from left to right and top to bottom, following the mouse cursor. The coordinates for the point where these two line cross are reported in the numeric cursor readout at the top of the graph area. Most commonly, we want to know the value of an actual data point and so the default behavior for the movable "crosshair" cursor, on all graph types except the spectrograph, is to follow the mouse cursor horizontally and snap to the nearest amplitude/magnitude data value on the vertical axis. If multiple traces are present the cursor tracks the front trace in the *z* order.

You can turn off data tracking for the movable cursor by unselecting *Free Cursor Tracks Data* in the *Command* > *Cursor* menu (a check mark appears on the left of the menu line when the option is selected) or by pressing [Ctrl/Cmd] + [Shift] + [F] on your keyboard. On the spectrograph of course, *both* the *x* and *y* axes represent absolute coordinates and so the movable cursor is always tracking actual data no matter where on the graph you put it.

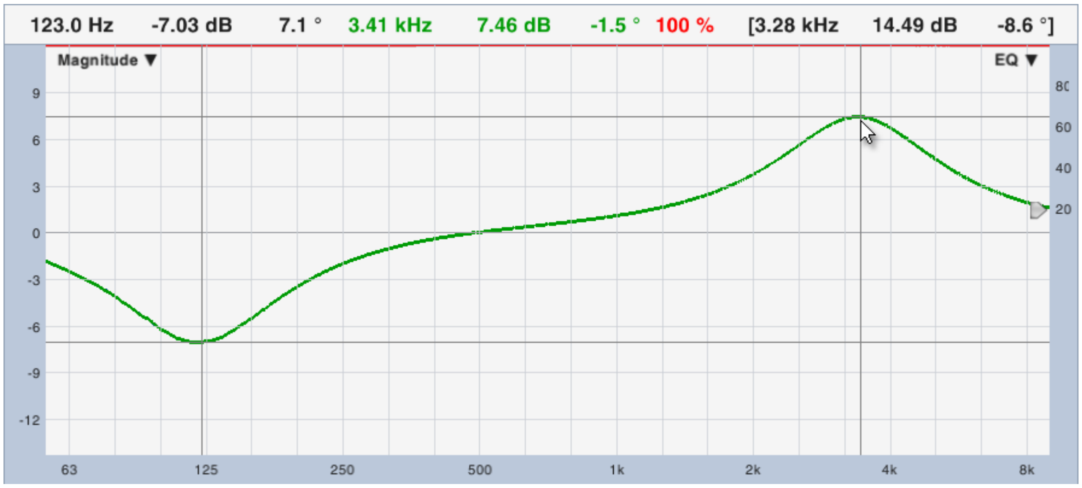


Figure 18: The cursor readout shows coordinates for locked and movable cursors.

Locked Cursors

In addition to the movable "free" cursor, you can also set a second cursor called a "locked" cursor on most plots in Smaart. The exceptions to this are spectrograph charts, the Live IR graph in real-time mode and the small navigation pane in impulse response mode. When a locked cursor is present, the cursor readout gives you coordinates for both cursors and displays the difference between the two in brackets.

To set a locked cursor, hold down the [Ctrl/Cmd] key on your keyboard while clicking some point on the plot that you are interested in, or press [Ctrl/Cmd] + [P] to set a locked cursor at the highest peak in the active plot. To remove a locked cursor press [Ctrl/Cmd] + [X].

If you turn on *Locked Cursor Tracks Data* in the *Command* > *Cursor* menu, or press [Ctrl/Cmd] + [Shift] + [L] on your keyboard the locked cursor will track the magnitude as it moves up and down. This option can be used in conjunction with the find peak function.

General Options

The *General* options dialog (*Options* menu > *General*) contains settings that apply to Smaart's appear-ance and behavior in both real-time and impulse response mode.

Line Thickness

The *Line Thickness* controls set the line thickness for all line charts in Smaart, including Transfer Function (Magnitude, Phase, Live IR) narrowband Spectrum (Linear/Log) and Impulse Response mode time- and frequency-domain displays.

- *Foreground Trace* – sets the line thickness in pixels for the data trace at the top of the *z* axis on all line charts.
- *Background Trace* – sets the line thickness in pixels for all data traces other than the top trace when multiple traces are displayed on any line chart.

Cursor Frequency Readout

Selections in this section determine what information appears in the cursor readouts for *y*-axis coordinates on frequency-domain graphs.

- *Frequency* – cursor readout displays only frequency in Hertz for the frequency coordinate corresponding to the mouse cursor position.
- *Frequency and Wavelength* – cursor readout displays frequency in Hertz and the wavelength in feet or meters, depending on the temperature and distance units selection in the *Speed of Sound* section below.
- *Frequency and Note ID* – cursor readout displays frequency in Hertz and the closest musical note corresponding to that frequency.

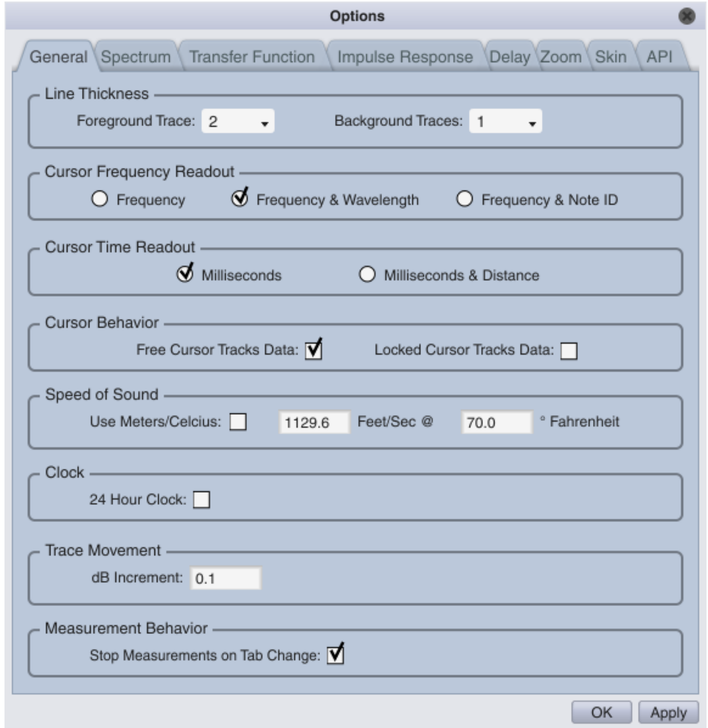


Figure 19: The General options page of the Options dialog

Cursor Time Readout

The cursor time readout setting applies to the both IR Mode and the Live IR graph on Transfer Function displays.

- *Milliseconds* – displays time coordinates and relative time differences in milliseconds only.
- *Milliseconds & Distance* – displays time coordinates as milliseconds and equivalent distance, based on the Speed of Sound settings (see below).

Speed of Sound

The settings in this section determine the speed of sound that Smaart uses for calculating equivalent distances for time coordinates, and whether distances are displayed in feet or meters. It can also serve as a handy speed of sound calculator any time you need to know the speed of sound for a given air temperature.

- *Use Meters/Celsius* – when this option is selected, Smaart displays distances in meters and the temperature used for calculating speed of sound in degrees Celsius. Otherwise, Smaart displays distances in feet and uses degrees Fahrenheit for temperature.
- *Speed of sound* ([unit]/sec) and temperature – At elevations where humans would be comfortable breathing, the speed of sound is mainly a function of temperature, and so these two input fields are linked. Changing the temperature setting automatically recalculates the corresponding speed of sound and vice versa.

Clock

24 Hour Clock – this option changes the clock display in Smaart from a 12-hour clock to 24-hour. The clock display replaces the numeric signal level/sound level meter at the top of the Control Bar when you select *Toggle SPL /Clock* in the *View* menu or press the [K] key on your keyboard.

Trace Movement

dB Increment – sets the move increment (in decibels) for the *Front Trace Up* (*keyboard shortcut:* [Ctrl/Cmd] + [↑]) and *Front Trace Down* ([Ctrl/Cmd] + [↓]) commands in the *Command* menu that move the front trace on the active graph up or down for display purposes.

Measurement Behavior

Stop Measurements on Tab Change – When this option is checked, Smaart automatically stops all running measurements on a tab when you switch to another tab in the same window. When unchecked, live measurements running on all tabs continue running until you stop them yourself or exit the program. The advantages of allowing background operation are that when you come back to a tab, you don't have to restart all of your measurements again and wait for their averages to populate. Also, if a remote client is subscribed to the window, then changing tabs on the host machine does not interrupt measurement data being streamed to the client. The caveat is that if you have a lot of measurements running on multiple tabs, a lot of processor cycles can get eaten up crunching data that you are not actually using and performance may suffer as a result.

The Signal Generator

Smaart's signal generator is a versatile and highly configurable signal source capable of driving all types of time- and frequency-domain measurements. It can generate effectively random pink noise, pseudo-random pink noise with optional band-limiting, speech-weighted noise, sinewaves, dual-sinewave signals, log swept sines (called "pink sweeps") and file-based signals from .wav or .aiff files. Some of the generator's more advanced options are particularly advantageous for acoustical impulse response measurements for room acoustics analysis and we will discuss those in more detail in the chapters pertaining to impulse response mode. For real-time measurements, we most often use pink noise in one form or another.

Signal generator controls appear on the Control Bar in both real-time and impulse response modes. These top-level controls enable you to select the basic signal type (*Pink Noise*, *Pink Sweep*, *Sine*, *Dual Sine* or *File*), adjust the output level, and turn the generator on and off. The output level readout is directly editable; you can click it with your mouse to edit, and then press the [Enter] key to set the new level. Signal generator options are accessible by selecting *Signal Generator* from the *Options* menu, pressing Alt/Option] + [N] on your keyboard, or by clicking on the heading on the *Signal Generator* section of the Control Bar, which becomes a button when your mouse cursor passes over it.

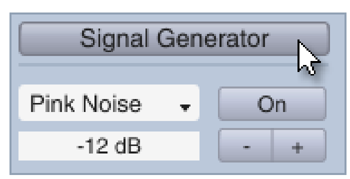


Figure 20: Signal generator controls on the Control Bar

In addition to the default control layout, a more compact layout is available for the signal generator by selecting *Compact Signal Generator* from the *View* menu. When this option is selected, the signal type is

Figure 21: Compact signal generator control layout

indicated on the button that turns the generator on and off, and clicking on the numeric level readout in the center opens the *Signal Generator* control panel.

Output *Device* and channel selections for the signal generator are set in the lower portion of the *Signal Generator* setup dialog. Smaart can route the output of the signal generator to any two outputs (*Main* and *Aux*) on any available audio output device. Note that all signals that Smaart generates are monaural. When an *Aux* output is assigned, the same signal is sent to both the *Main* and *Aux* channels. If you specify a stereo .wav or .aiff file for file-based signal generation, Smaart uses only the left channel and ignores the right channel.

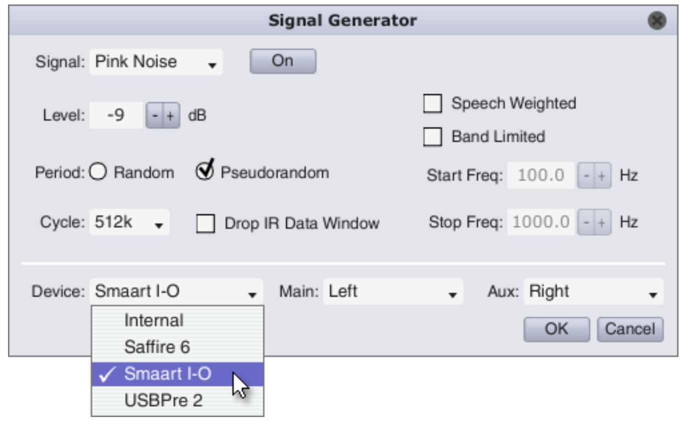


Figure 22: Signal generator output device and channel selection

The upper portion of the *Signal Generator* dialog is devoted to selection and configuration of the output signal type. These options vary somewhat depending on the type of signal currently selected on the *Signal* selector in the upper left corner of the dialog window.

Pink Noise

Smaart can generate two basic types of noise, which we refer to as *Random* or *Pseudorandom*. *Random* pink noise is created by streaming the output of a random number generator through a digital filter network, much the same way that most hardware pink noise generators work. Technically, the random number generator is pseudorandom also, however it is randomly reseeded every time you start the generator and given its cycle length of 2^{19937} samples, it will effectively never repeat.

Pseudorandom noise signals in Smaart repeat on intervals that are a power-of-two samples in length up to 2^{19} (512K samples). These can be band-limited or shaped to an idealized long-term average speech spectrum (*LTASS*) in addition to broadband pink noise. When using pseudorandom noise, you should always select a cycle length that is at least as long as the longest FFT size that you are using for measurements. The longest time window used in the MTW transfer function is a little over one second, so 64K would be the lower limit in that case but noise sequences even that short tend to become hard to listen to pretty quickly.

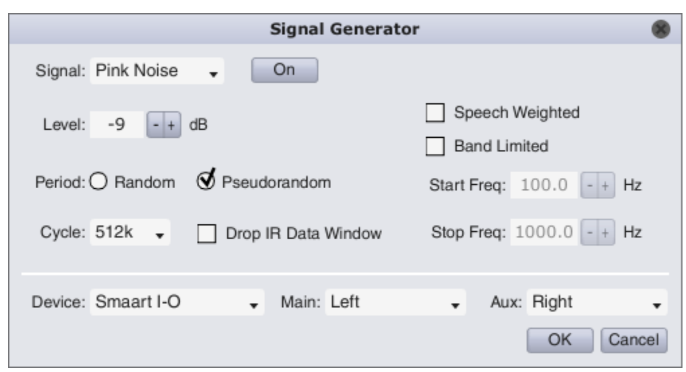


Figure 23: Signal generator options for "Pink Noise" and other pseudorandom shaped noise signals

For real-time measurements in general, a setting of 512K or 1024K is generally a good choice for *Cycle* time. 512K works out to about 11-12 seconds at 48 kHz sampling rates, which is long enough to make repeats unnoticeable but still short enough for averaged measurements to settle relatively quickly. You may want to increase the size for higher sample rates. For impulse response measurements in IR mode, the *Drop IR Data Window* option automatically sets the sequence length to match the FFT size for dual-channel IR measurements, so that they can be recorded without a data window.

There are three spectral options for pseudorandom noise: broadband pink, band-limited pink and speech-weighted. Pink noise has a spectrum that appears flat on a fractional-octave RTA display and rolls off at 3 dB per octave (10 dB per decade) on a narrowband frequency scale. Band limited pink noise is a signal with a nominally pink spectrum across a specified bandwidth. Checking the *Band Limited* option activates the *Start Freq* and *Stop Freq* controls enabling you to specify your desired passband. *Speech Weighted* noise is pseudorandom noise with a spectral shape based on the idealized long-term average speech spectrum (LTASS) defined in ANSI S3.5-1997.

Digital signal levels for all noise signals are calibrated to normalized full scale, which is to say that the maximum possible amplitude for a given sample word size (e.g., 24 bits) is equal to 0 dB peak. Random and pseudorandom pink noise signals are hard limited to ensure a peak-to-RMS ratio of 12 dB.

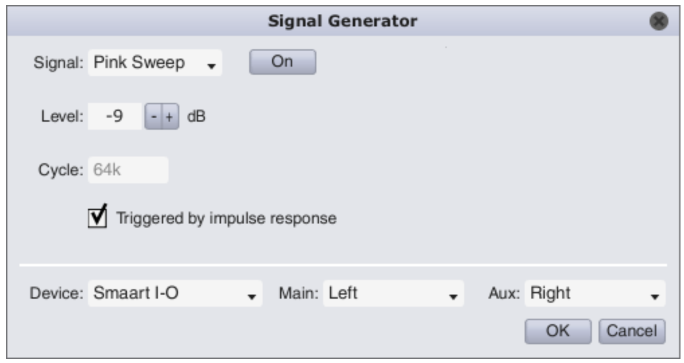


Figure 24: Signal generator options for Pink Sweep (log-swept sine) signals

Pink Sweep

The *Pink Sweep* signal is a logarithmic sinusoidal sweep intended primarily for use in impulse response measurements for room acoustics analysis. A sweep signal consists of a short period of silence, followed by the sweep sequence, followed by another period of silence. The sweep itself takes up just half of the selected cycle period.

In impulse response mode you can use the *Triggered by impulse response* option to automatically set the sweep length to match the FFT size used for the IR measurement and trigger the generator automatically when you start a measurement. Note that you will generally want to measure and set the delay time between the measurement and reference signals measurements using sweeps unless the expected delay time is very short, and typically little or no averaging is necessary or even desirable.

Sine and Dual Sine Waves

The options for *Sine* waves and *Dual Sine* signals are essentially identical. Selecting *Sine* just removes the bottom row of controls that you see in Figure 24. The relative signal level for each sinewave (Level 1 and Level 2) is set independently and the master Level control controls the overall signal level.

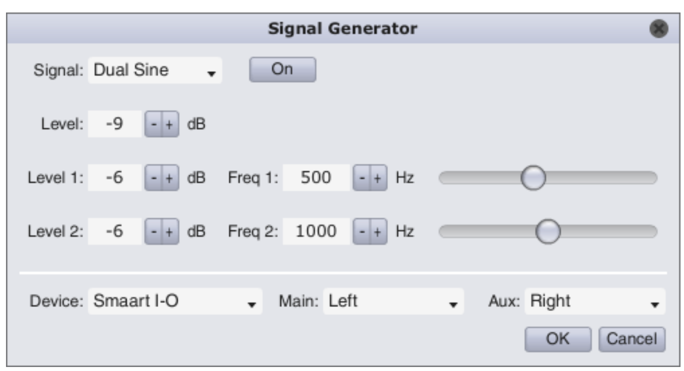


Figure 25: Signal generator options for Dual sinewave signals

Note that Smaart does not use the AES convention for digital Full Scale when stating output levels for test signals. Signal levels for all generator output signals are stated relative to normalized full scale peak, meaning that the maximum peak amplitude of a full scale sine wave is 0 dB (not +3 dB) and the maximum RMS level is -3 dB (not 0 dB).

File-based Signals

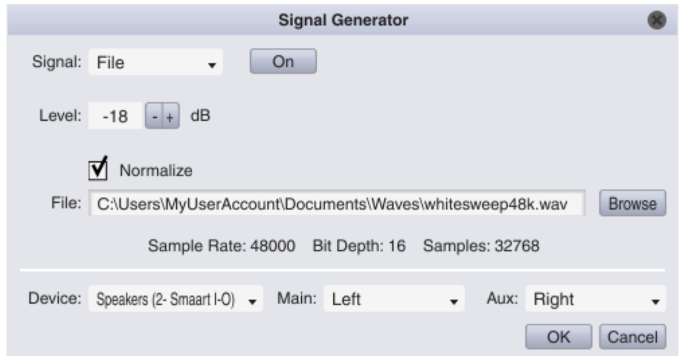


Figure 26: Signal generator options for file based signals

In addition to internally generated signals, Smaart also enables you to use any .wav or .aiff file as a test signal. When using file-based signals, you just need to select the file you want to use and specify the output level. When the *Normalize* option is selected, Smaart will scale the signal to a peak level of 0 dB normalized full scale. Note that Smaart's signal generator always sends the identical signal to both the *Main* and *Aux* output channel selections. If the source file is stereo, only the left channel is used and the right channel is ignored. Also be aware that Smaart copies the entire file into RAM to provide seamless looping, so you may want to keep your file lengths fairly short.

The Command Bar

The Command Bar is a user-configurable button bar that runs across the bottom of a Smaart window. It can be hidden and restored by means of the triangular button centered in the border area just above it. The show/hide button remains visible in the window border when the Command Bar is hidden and clicking this button again will restore it. You can also hide or restore the Command Bar by selecting *Command Bar* in the *View* menu or by pressing the [U] key on your keyboard.

Configuring the Command Bar

The Command Bar consists of ten buttons, any of which can be assigned any one of a wide range of functions. In general, nearly anything you can do with a keyboard shortcut can also be done with a command bar button. Command Bar button functions are assigned in the *Command Bar Configuration* dialog, accessible by selecting *Command Bar Config* from the *Config* menu.

To assign a function to a Command Bar button, click on one of the selectors in the *Command* column on the right and select the function that you want to assign from the list. A default name for the function will be automatically suggested in the corresponding *Name* field on the left. You can click on this field to make it editable if you want to change the name. After editing a button name, press the [Enter] key to set the change. When you are finished configuring your selections, click *OK* to exit the dialog. You should immediately see your new button names on the Command Bar.

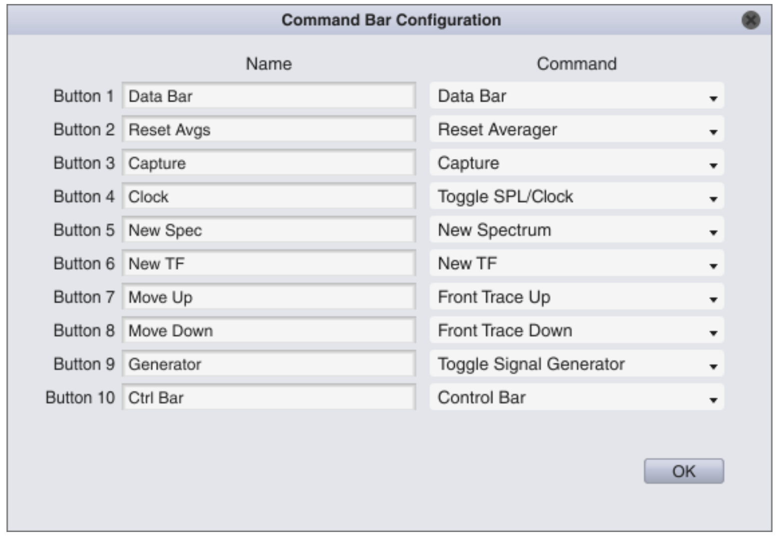


Figure 27: The Command Bar Configuration dialog

Input Meters Window

The Input Meters window displays a graphical peak-reading signal level meter calibrated to normalized full scale for each input channel that is currently selected for use in Smaart on the *I-O Config* tab of the *Configurator* dialog. This window is accessible by selecting *Input Meters* > *Input Meters Window* from the *View Menu* or pressing [Shift] + [E] on your keyboard. Repeating either of the actions while the window is open will close it.

Vertical Meter Orientation

Horizontal Meter Orientation

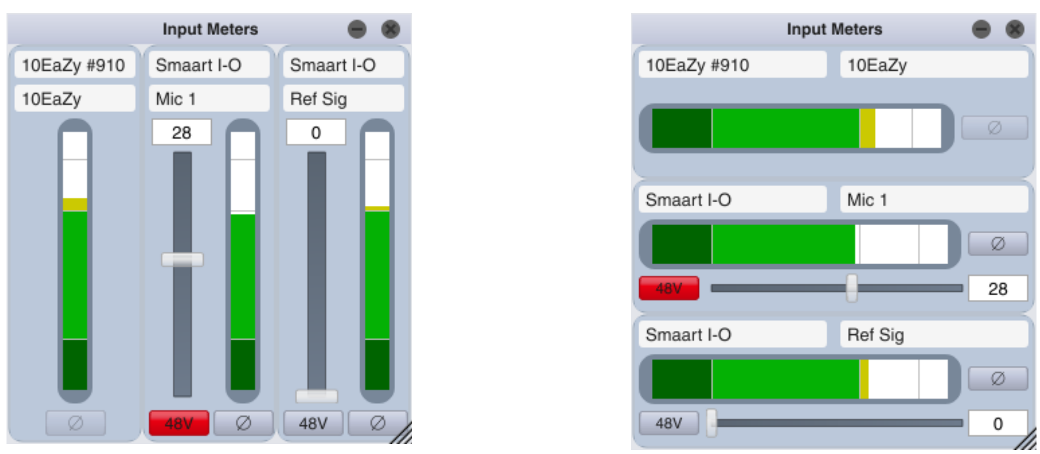


Figure 28: Input Meters window with horizontal and vertical orientation

The *Input Meters* window is resizable and its component meter modules scale with window size, meaning that you can expand the window to display larger meters or shrink the window and all of its contents to take up less space or fit more meters on your screen. You also have the option of a horizontal or vertical window layout. To change the orientation of the *Input Meters* window, you can select *Input Meter* > *Input Meters Orientation* from the *View* menu or press [Alt/Option] + [Shift] + [E] on your keyboard. Repeating either action will change it back.

The individual level meters that make up the Input Meters window are multicolor, with each color representing a specific amplitude range. Signal levels below -60 dB FS Peak are shown in dark green, transitioning to lighter green from -60 to -12 dB, then yellow from -12 and -6 dB and red above -6. Since the meters are calibrated to normalized full scale peak, the maximum possible value is 0 dB. If a clip is detected on any input channel, its entire meter bar turns red until the condition is resolved.

Each meter module includes a polarity invert button (useful if you have a microphone that's wired Pin 3 +) and level meters for Smaart I-O inputs also include input gain and 48V phantom power controls. Note that invert buttons for 10EaZy devices are disabled since their microphones have known polarity.

Sound Level (SPL and L_{eq}) Metering, Monitoring and Logging

Smaart offers multiple means for monitoring signal levels, sound pressure levels (SPL) or equivalent sound levels (Leq). These include an in-Tab sound level meter in the main window that is dynamically assignable to any configured input channel on the fly, a separate, stand-alone *SPL Meters* window containing some number of sound level meter modules assigned to specific input channels, and an *SPL History* window that works with Smaart's sound level logging functions to provide a graphical view of changes in sound levels over time.

For any calibrated input being metered or logged, Smaart simultaneously measures 10 built-in sound level metrics along with digital full scale peak and user-defined Leq measurements (if applicable). Built in metrics include standard *Fast* and *Slow* sound pressure levels (SPL) and equivalent sound level (Leq) measurements with a one-minute integration period. Each is available with standard *A* or *C* frequency weighting or unweighted, for a total of 9 separate SPL and Leq metrics. Additionally, *C*-weighted peak levels (*Peak C*) are also calculated for each calibrated input channel assigned to a meter or a log file along with unweighted peak levels referenced to normalized digital full scale. Simultaneous calculation of all metrics means that you can switch any meter between measurements types for a given input instantly – without waiting for average buffers to repopulate – and log *all* available measurement types to an ASCII text files for record-keeping purposes and off-line analysis.

Sound Level Metering

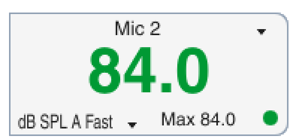


Figure 29: Main Window in-tab SPL meter pane

The large numeric readout that appears (by default) at the top of the Control Bar, in the upper right corner of each tab in main Smaart program windows can be configured to function as a sound pressure level (SPL) meter, an integrating equivalent sound level (Leq) meter, or a peak signal level meter calibrated to normalized digital full-scale for a single input channel. If the meter pane is hidden, you can restore it by selecting *SPL Meter* in the lower section of the *View* menu or pressing [Alt/Option] + [K]. When visible, it can also be replaced by a clock by selecting *Toggle SPL/Clock* from the *View* menu or pressing the [K] key on your keyboard.

In addition to the in-Tab SPL meter, Smaart can also display signal levels and sound levels for multiple inputs in a separate *SPL Meters* window. The *SPL Meters* window is composed of some number of meter modules arranged in a grid. It must be configured in the *SPL Config* page of the *Configurator* before it can be used. *SPL Config* is covered in detail in the next chapter, beginning on page 61. Once configured, the SPL Meters window is accessible by selecting *SPL Meters* from the *View* menu or by pressing the [E] key on your keyboard. The *SPL Meters* window is resizable and its meter modules scale with window size, meaning that you can expand the window to display larger meters or shrink the window and all of its contents to take up less space or fit more meters on your screen.

In-tab SPL meters and individual meter modules in the *SPL Meters* window operate nearly identically. The only functional difference between them (other than the clock option) is that the in-tab meter can be switched from one input to another directly from the meter display – the top line of text in the meter pane showing the name of the input channel currently assigned to the meter also functions as a menu for selecting the input when you click on it with your mouse. Meter names and input assignments for meter modules in the *SPL Meters* window are assigned in *SPL Config* tab of the *Configurator* dialog and can only be changed from there. Otherwise, there are some small differences in layout but both have the same basic components and operate the same way.

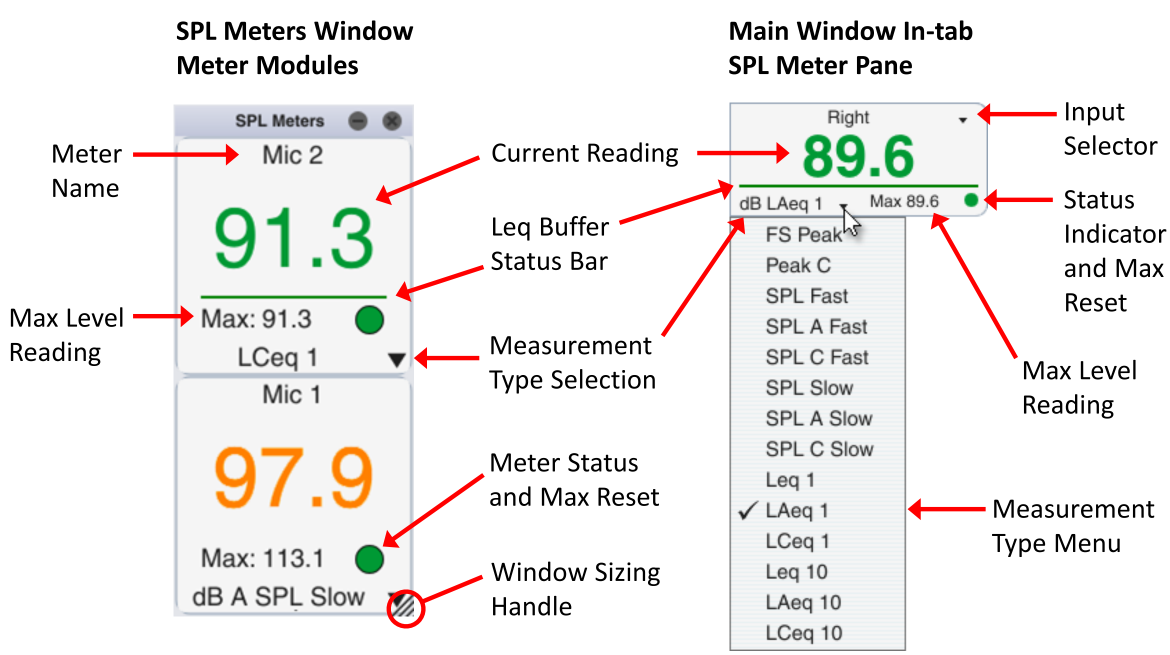


Figure 30: SPL Meters window with a 1x2 (one module wide, two modules high) grid layout, shown alongside a main window SPL Meter pane with its measurement type menu expanded.

The large number in the center of the meter is of course the current signal level or sound level. On SPL and Leq meters, the text color of this number can vary with sound level, according to the specified *Green Above*, *Yellow Above*, and *Red Above* threshold levels specified in *SPL Config*. Below the specified *Green Above* level, the number changes to the default SPL meter text color for the current color scheme. Full Scale signal levels are always displayed in the default text color.

Below the current reading, the measurement type selection and maximum (*Max*) reading recorded since metering began or since the last reset are shown. In the *SPL Meters* window, these are stacked on two lines. On an in-tab meter they appear side-by side.

To the right of the *Max* value is a colored circle that functions as a status indicator for the meter and also as a reset button for the *Max* reading. It is green when the meter is happy and turns red if the input is overloaded. The *Max* reading will change to read "OVERLOAD" in that case also. Status indicators on SPL and Leq meters will also turn gray if the signal level falls below the *"Green Above"* level specified in *SPL Config*. Additionally, the border of the *SPL Meters* window will flash red if alarm levels defined in *SPL Config* are exceeded on either of two specified input channels. Clicking on the indicator with your mouse clears the previous *Max* reading.

Clicking the measurement type on any meter with your mouse will pop up a menu of all measurement types available for the input channel currently assigned to the meter. If the input is calibrated for sound level measurement, you will have your choice of eleven built-in measurement types plus any user-defined Leq measurements that you have configured. If not, then the only option available will be *FS Peak,* which sets the display to a peak reading signal level meter calibrated to normalized full scale.

Note that two groups of Leq measurements appear in the measurement type menu shown in *Figure 30*, Leq 1 and Leq 10 (with *A*-weighted, C-weighted, and unweighted versions of each). The one-minute Leq measurements are built-in, the others are user defined. User-defined Leq settings are configured from the *Leq Config* dialog, accessible by clicking the *Leq Config* button under the meters table on the *SPL Config* tab of the *Configurator* dialog.

When the measurement type assigned to a meter is an Leq measurement, a buffer status bar appears immediately below the large current level reading in the center of the meter. When you begin a measurement or reset the Leq buffers for the selected input, the indicator starts out as a short orange line segment on the left that gradually lengthens as the average accrues. When the averaging buffer is fully populated, the line changes from yellow to green and from that point on, Smaart will begin continually removing the oldest data from the Leq average as new data arrives to replace it.

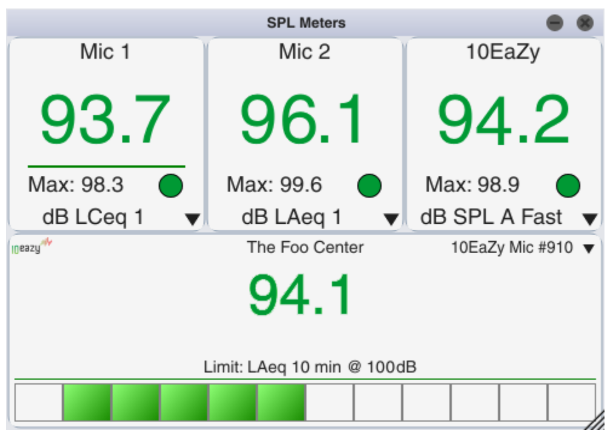


Figure 31: A 1x3 SPL Meters window with 10EaZy MAM display appended

Clicking anywhere in the center portion of any meter opens the *Configurator* dialog to the *SPL Config* page. Right-clicking on any meter pops up a menu enabling you to reset the Leq buffers for the selected input. On an in-tab meter, you can also toggle the meter pane between a meter or a clock, or access calibration settings for the selected input from the context menu.

10EaZy Maximum Average Manager (MAM)

If you have one or more SGAudio Aps 10EaZy devices connected to your computer, a 10EaZy MAM display can be appended to the SPL Meters window as shown in the figure on the right. The MAM is a predictive algorithm designed to help live sound engineers keep the output of a sound system within a specified sound level limit, specified in terms of equivalent sound level (Leq) such as Leq 10 or Leq 15. If you are familiar with the MAM display as implemented in the 10EaZy's bundled software application, the MAM display in Smaart should require little explanation because they work identically.

The large number in the center of the display is the current Leq level for the microphone being monitored. Note that Leq settings for the MAM display are specified independently of those for all other meters in Smaart. Below the current reading, the Leq integration period and limit level for the MAM are shown. These are defined in the *10EaZy Maximum Average Manager Config* section of *SPL Config*.

The graphical “LED” bar at the bottom of the MAM display shows you how much headroom you have at current sound levels or how much you will exceed your target level if current sound levels continue. Each segment on the bar represents 1 dB Leq. Green segments to the left of center indicate how much louder the sound level could be without exceeding your limit. Red segments to the right of center predict that the target level will be exceeded and show how many dB the current level would need to be reduced to achieve the specified limit level. In general, this display is most meaningful after about 30 seconds at representative sound levels and may be misleading during breaks between songs or other pauses in the audio program.

The *Venue Name* specified in *SPL Config* is normally shown at the top of the MAM display, above the current Leq reading and the input name of the device being monitored appears in the upper right corner. The name shown here is the *Friendly Name* specified for the input channel on the *I-O Config* tab of the *Configurator* dialog. If you have more than one 10EaZy connected, you can switch between devices by clicking on the input name and selecting a different device from the pop-up menu. Note that if you shrink the *SPL Meters* window below the point where there is room for both, the venue name moves to the right corner, replacing the input name. You would then need to expand the window in order to see both.

SPL History Window

The SPL History window works with Smaart's sound level logging functions to provide a graphical display of sound level measurements over some period of time. It can be used to monitor live measurements currently in progress or as a file browser for previously recorded data stored in sound level log files. This window is accessible by selecting *SPL History* from the *View* menu, pressing [Alt/Option] + [H] on your keyboard, or by clicking the SPL History button on the *SPL Config* tab of the *Configurator*.

The *SPL History* window consists primarily of one or two graph panes. When you initially open the window, both graphs will be blank until you have selected a data source. If sound level logging is turned on, the *Source* selection can be either a live input or a log file. Note that only inputs currently selected for logging in *Log Config* are available as live data sources in this case (see *Sound Level Logging* on page 65 for details). When logging is turned off, the *Source* selection automatically defaults to *File* and the *File Path* field becomes active. You can click the *Browse* button to the right of the *File Path* field to navigate to the log file that you want to display.

When you have selected a data source, sound level data from the source is plotted on the two main graphs. The upper of the two panes is always visible and functions as an overview, showing all data in the file or all data collected since logging began for a single measurement type (SPL, Leq or Peak C).

The (optional) lower graph pane can display any combination of the 13 available measurement types for a specified time range, which can be the entire time record or any portion thereof. If you have *Alarm* levels set up in *SPL Config* for a displayed measurement type, these are plotted on the graph as well.

The time range for the lower graph is selected by clicking and dragging the two arrowhead-shaped widgets in the overview graph pane to the left or right with your mouse.

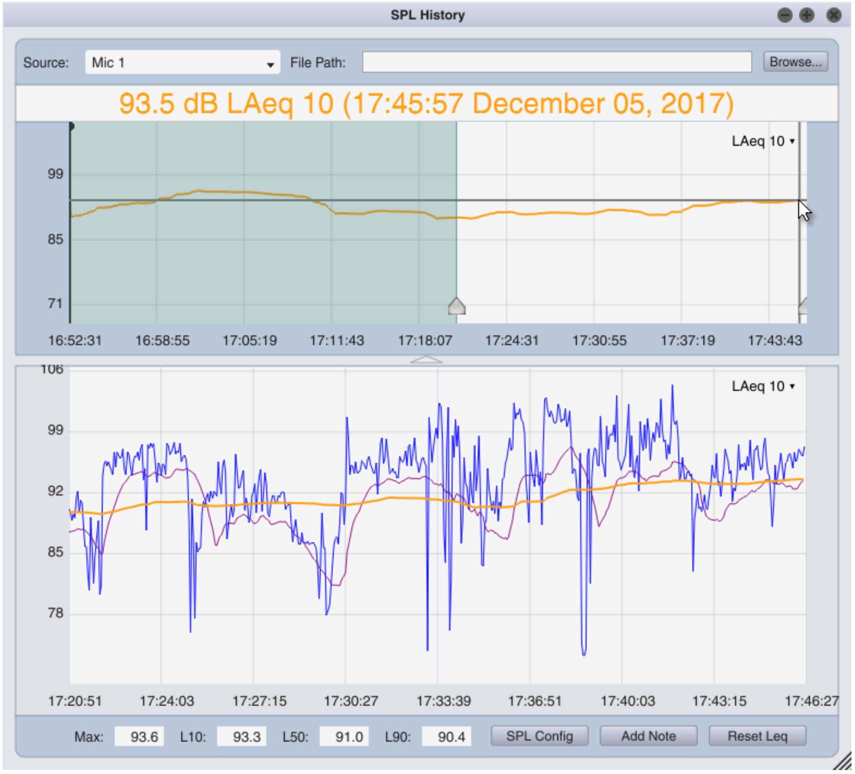


Figure 32: The SPL History window

The measurement type displayed in the upper graph will be the front trace on the main graph below. The name of this measurement is shown in the upper right corner of each graph. Clicking on the name of the front trace displays a list of all available measurement types. You can click on the boxes to the left of each measurement type to select the measurements that you want to appear on your graphs. Clicking on the name of a measurement that is selected for display will bring it to the front of the graph. You can also use the [Z] key to shuffle the z-axis stacking order as you can with most other graphs in Smaart.

Below the history graphs are some statistics about the front trace including maximum (*Max*) reading recorded for the currently selected time period and three additional figures labeled *L10*, *L50* and *L90*. These refer to percentile rankings of all the readings recorded within the designated time range. All of these readings are stated in decibels.

- *L10* refers to the loudest 10% of all the readings recorded – in other words, 90% of all individual readings taken were below this level.
- *L50* is the median reading – half of all readings recorded were less than or equal to this level and half were higher.
- *L90* is the level that was exceeded 90% of the time and is generally regarded as a useful estimation of background noise levels.

The *SPL Config* button provides a direct link to the *SPL Config* page of the *Configurator* dialog, where most controls for sound level metering and logging are located. *SPL Config* is covered in detail in the next chapter, beginning on page 61.

The *Add Note* and *Reset Leq* buttons are available when a live input is selected as the data source for the SPL History graphs. Clicking the *Add Note* button pops up a dialog window for adding a time-stamped note to the header of the log file currently being recorded. *Reset Leq* flushes the averaging buffers for all Leq measurements associated with the selected input and restarts the averages.

When you reset the Leq buffers for a logged input channel, a black "lollipop" marker (a vertical line with a circle on top) appears on graph to indicate the time position of the reset. Note that same thing happens automatically any time the input being logged is overloaded, with the time position of the clip indicated on graphs by a *red* lollipop marker in that case.

Color Schemes

Smaart has two built-in color schemes, the Default Dark (light-on-dark) scheme that you see the first time you run the program and a Default Light scheme with darker-colored text and data traces on a light background. The Default Dark scheme works well for indoor work, particularly in darkened rooms. The Default Light scheme may be a better choice if you are working outdoors in daylight or in a brightly lit room, or when making screen shots for printed documents or slide presentations. Most of the screen captures in this document were made using the Default Light color scheme.

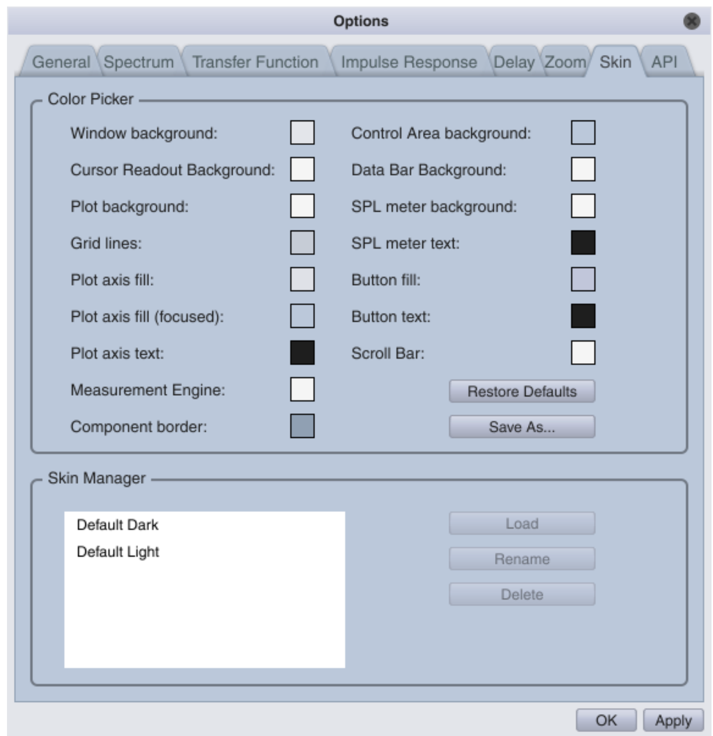


Figure 33: The Skin tab of the Options dialog

To switch to the high contrast color scheme, you can either select *Default Light* from the *View* > *Skins* submenu or press [Ctrl/Cmd] + [Shift] + [X] on your keyboard to cycle through all available color scheme. Note that a check mark appears in the next to the name of the currently selected color scheme in *View* > *Skins* submenu.

Custom Color schemes

In addition to the built-in default color schemes, Smaart also enables you to define custom color schemes or “skins” of your own. To define a custom color scheme, select *Skin Manager* from the *Options* menu. This opens the *Options* dialog to the *Skin* page.

The *Skin* options page is divided into two sections, *Color Picker* and *Skin Manager*. The *Color Picker* section consists of a number of color tiles showing the current selections for various elements of the current Smaart color scheme. Each color tile is a button that opens a *Color Selection* dialog, wherein you can specify a color for the associated GUI element. Color changes take effect immediately in Smaart when you click the *Apply* button in the *Color Selection* dialog to apply your change and exit the dialog.

When you have configured a color scheme to your liking, you can click the *Save As* button and give it a name to save it to the list below. Current color selections are part of your current program configuration but named color schemes are saved separately and are available to all configurations. The *Restore Defaults* button restores the *Default Dark* color scheme.

The *Skin Manager* section of the *Skin* options page has a list box on the left side listing all available named color schemes. Clicking on the name of a color scheme in the list and then clicking the *Load* button applies its settings to Smaart's current color selections. Note that you can also cycle through your configured skins from a top-level Smaart window, without opening the skin manager, by pressing [Ctrl/Cmd] + [Shift] + [X] on your keyboard.

The built-in *Default Light* and *Default Dark* color schemes cannot be deleted or renamed but they can be used as the basis for your own custom skins. Simply select one or the other and click the *Load* button, then make your modifications in the *Color Picker* section above and click the *Save As* button. The *Rename* and *Delete* buttons in the *Skin Manager* section can be used to delete or rename custom color schemes.

Managing Configurations

Nearly all of Smaart's user-configurable settings are stored in an XML file on shutdown, so that Smaart can "remember" them the next time you run the program. Configuration can also provide an easy way to switch between setups for different tasks or work environments. Configuration (config) files are stored in the *Config* subdirectory of the *Smaart v8* folder in the *Documents* folder for your user account. Named configurations can be created and managed through the *Config Management* dialog, accessible from the *Config* menu (*Config* > *Manage Configurations*). Copies of config files for a given computer can be saved to another location for backup purposes if you like, however we do not recommend attempting to move them to another machine, due to the environment-specific information they contain.

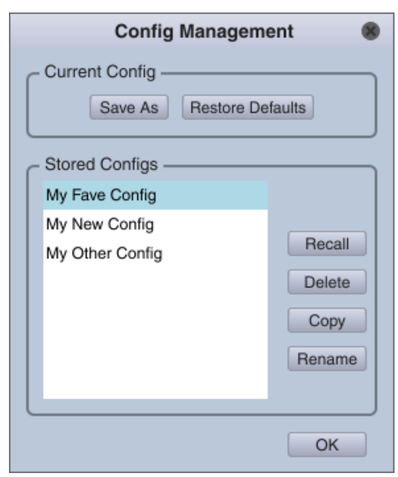


Figure 34: The Config Management dialog window

The current configuration includes the names, layouts and locations all of your windows and tabs, audio device selections including friendly names and calibration settings, measurement setups for each tab, view presets, and all but a handful of menu and dialog settings. It is updated each time you make a settings change in Smaart, so once you have Smaart set up to your liking for a particular scenario, saving a copy of the current configuration will enable you to return the program to that setup in the future. To store a configuration, open the *Configuration Management* dialog and click the *Save As* button in the *Current Config* section. You will be prompted to name the new configuration. When you have done so, click OK to close the pop-up window and your new configuration will appear in the *Stored Configs* list.

To make a copy of a stored configuration select its name and click the *Copy* button. Here again, you will be prompted to name the new config file. To delete a stored configuration that you no longer need, select its name and click the *Delete* button.

Chapter 3: Configuring Smaart for Real-Time Measurements

Before you can do much of anything with Smaart, you will need to configure one or more live measurement objects. We refer to these variously as measurement engines, measurement objects, or just plain measurements. Each live spectrum or transfer function measurement in Smaart is essentially a complete real-time spectrum or transfer function analyzer. They can be driven by any audio streams that your computer has access to and there is no set limit on how many that you can create and run simultaneously. There may be a *practical* limit, depending on such factors as available computing resources (RAM, CPU, and GPU) or how many microphones and input channels you can afford, but Smaart imposes no limits of its own.

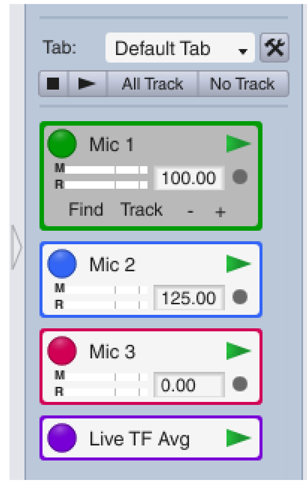


Figure 35: Live transfer function measurement control blocks on the Real-Time mode Control Bar

Spectrum and Transfer Function Measurements

The first step in setting up spectrum or transfer function measurements is to choose the input devices and channels that you want to use with Smaart. This is done from the *I-O Config* page of the *Configurator*, accessible by selecting *I-O Config* from the *Config* menu or using the keyboard shortcut [Alt/Option] + [A]. When you start up Smaart for the first time or with a new configuration, it will pop up a message saying you have no *measurements* configured, and clicking *OK* in the message dialog will take you directly to *I-O Config*. The *Configurator* will automatically create spectrum measurements for each input channel on each input device that you select for use, with their user-specifiable "friendly names" as the measurement names.

Once you have made your input selections, you can create transfer function measurements using those inputs as well, either from the *Measurement Config* page of the *Configurator* or by selecting *New TF* from the *Config* menu (*keyboard shortcut:* [Ctrl/Cmd]+[T]). You can also create additional spectrum measurements from the same inputs if you have some reason to do so, either in *Measurement Config* or by selecting *New Spectrum* from the *Config* menu, or you can press [Ctrl/Cmd]+[S] on your keyboard. When you create a new measurement, a control block for it is added to the Control Bar.

Creating a new measurement can be as simple as typing a name and selecting an input device and channel(s) to drive it. Smaart will assign a trace color automatically and the default settings for spectrum and transfer function measurements generally work well for most applications. Spectrum measurements need only a single input channel. For transfer function measurements you need two; a measurement signal (abbreviated *"Mea Ch"* in Figure 36) that is the output from a device or system under test, along with its input signal as a reference (*Ref Ch*) signal.

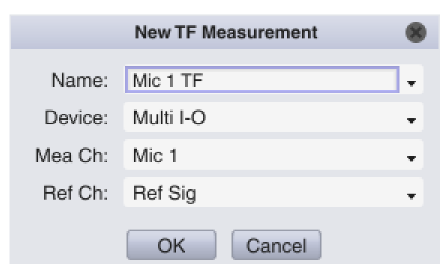


Figure 36: The New TF Measurement dialog

Note that all measurements on any given tab must be uniquely named; however, different tabs can have measurements with the same names as measurements on other tabs. If you want a measurement to appear in more than one tab or window, you can create copies by dragging and dropping in the tree view in *Measurement Config.* Another way to do it is to open one of the new measurement dialogs (see previous page) and instead of typing a name, click the down arrow next to the *Name* field to see a list of existing measurements, and then select the one that you want. Identically named measurements share a common display color and input channel selections and any changes to these settings are applied to all copies. Other measurement settings can be set independently for each copy.

Live Averages

Live averages are measurements calculated by averaging the output of other spectrum or transfer function measurements in the group. They are used primarily for real-time spatial averaging of measurements taken from multiple microphone positions.

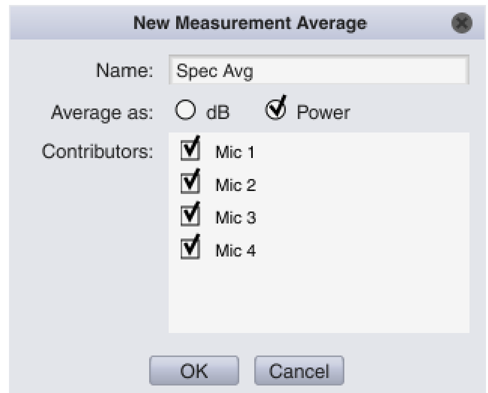


Figure 37: New Measurement Average dialog for a live spectrum average

To create a new live average, select *New Spectrum Avg* or *New TF Avg* from the *Config* menu or click the *New Spectrum Average* or *New TF Average* buttons on a tab view in *Measurement Config*. Either action will open the *New Measurement Average* dialog for the corresponding measurement type. In the dialog you will see a list of measurements of the same type on the current tab with a check box next to each one. Type a name for your new live average then click the check boxes for the measurements that you want to include in the average.

If you are creating a new spectrum average, you will also be asked to select *Power* or Decibel (*dB*) averaging. For transfer function live averages, you also have the option of using coherence weighing for dB averaging in this case. (You can change these settings later if you wish in *Measurement Config*.) Note that transfer function power averages are automatically level-adjusted to prevent any one measurement from dominating the average too much. Spectrum power averages are not, since this would not be desirable in all cases.

Decibel (*dB*) averaging is a simple arithmetic average of decibel magnitude values, giving equal weight to all frequency data points in all measurements going into the average. Coherence weighting in transfer function averages gives more weight to the frequencies in each measurement that have the highest coherence values. Coherence tends to be a predictor of a signal-to-noise ratio in transfer function measurements and so higher coherence suggests that the data may be more trustworthy. For more details on spatial averaging options, please refer to *Spatial Averaging,* beginning on page 19.

Audio I-O Configuration

The *I-O Config* page of the *Configurator* dialog is where you go to select and configure audio devices in Smaart. It is directly accessible by selecting *I-O Config* from the *Config* menu or using the keyboard shortcut [Alt/Option] + [A]. Here you will select which devices and channels that you want to see in lists, assign meaningful "friendly names" to your inputs and outputs, calibrate inputs, and apply microphone correction curves.

The S*ample Rate* and sample word size (*Bits per Sample*) controls in the *Global Settings* section apply to all input and output channels. The rest of the page is devoted to the devices table on the upper left and the channels table for the selected device below.

Configuring Input and Output Devices

At the top of the *I-O Config* page on the left are two buttons for *Input Devices* and *Output Devices* (see figure below). These select which type of devices are shown in the devices table. Of course, most physical audio devices have both inputs *and* outputs, but the operating system conceptualizes their inputs and outputs as belonging to separate virtual "devices." The devices table below the selector buttons lists all audio devices of the selected type that your operating system knows about. This table is comprised of four columns, labeled *Use*, *API: Driver Name*, *Friendly Name*, and *Status*. The layout of the table is the same for both input and output devices.

- To select a device for use in Smaart, click its check box in the *Use* column.
- The *API : Driver Name* column lists the name that the device or its driver reports to the operating system. On Mac OS X, all devices will use the CoreAudio API. Windows machines may have ASIO and Wave API devices and some devices may appear as both types. If you have both Wave and ASIO drivers installed for an I-O device with more than two channels, it will typically show up as one ASIO device and multiple Wave devices, because the wave API supports only two input channels per device.
- Clicking on any entry in the *Friendly Name* column makes the name editable so that you can type in whatever you want the name to be. Press the [Enter] key after editing, to apply your change.
- *Status* – The status for a device can be "*OK*," meaning that Smaart was able to connect to it successfully at start-up time, or "*N/C"* (not connected). *N/C* can mean *really* not connected – Smaart remembers audio devices that it has seen before, even when they are not present – or if the device is present and connected, it could indicate that some hardware or software problem prevented Smaart from communicating with it on start-up. This could be a hung device driver, a loose cable, or perhaps the device has become unresponsive and needs a reboot. In that case, you will likely need to restart Smaart once the problem has been corrected in order to see the device as *OK* and ready to use. You can remove an *N/C* device by selecting it in the list and clicking the *Remove* button below the devices table.

Configuring Input and Output Channels

Below the devices table is another table listing the individual channels for the selected device. Notice that a new tab is added to this table for each device that you designate for use in the table above. In this case, the layout of the table depends on the device type selection (*Input Devices* or *Output Devices*) at the top of the page, but the only user-specifiable option for output channels is the *Friendly Name* which we have already talked about.

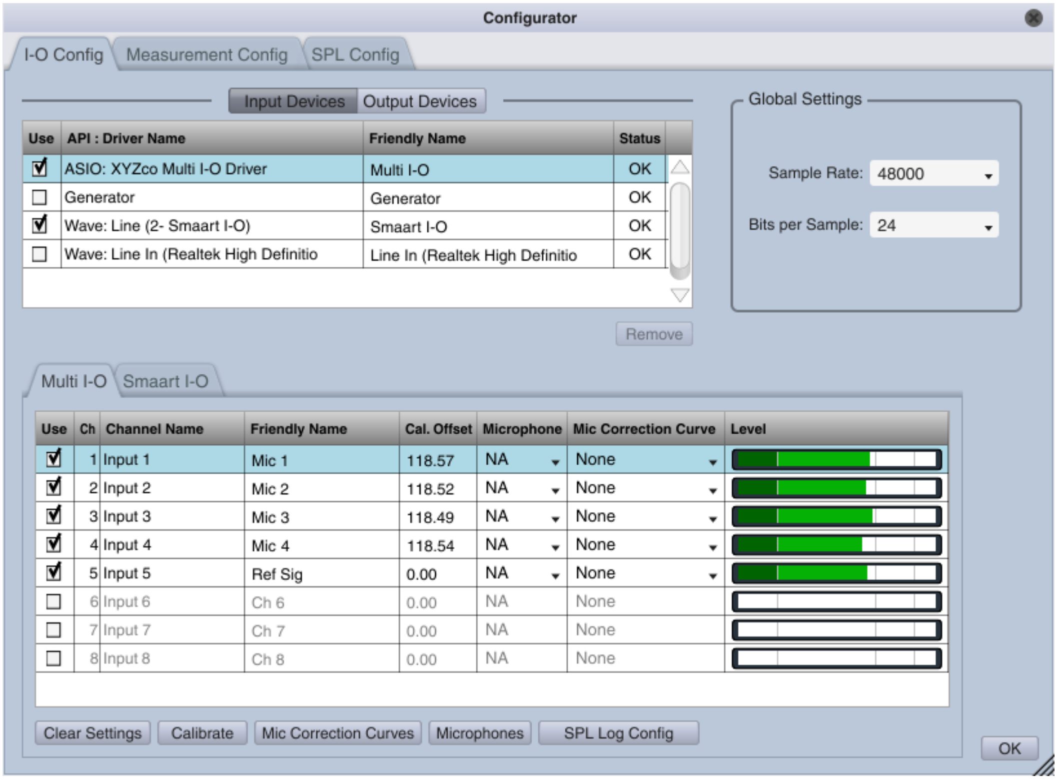


Figure 38: The I-O Config page of the Configurator dialog

The input channels table has eight columns, the last of which is a live signal level meter for each input.

- The check boxes in the *Use* column work the same as in the devices table. You can select or un-select the channels that you want to use or ignore by clicking their check boxes.
- The channel (*Ch*) column lists the channels by number.
- The *Channel Name* column lists the official (driver-reported) channel names. These are not editable.
- Clicking on any entry in the *Friendly Name* column makes the name editable so that you can type in whatever you want the name to be. Press the [Enter] key after editing, to apply your change and move on down the list to the next channel.
- The numbers in the *Cal. Offset* column represent the difference in decibels between each input channel's full-scale digital amplitude and its calibrated level. A calibration offset of zero means the input is calibrated to digital full scale. When an input is calibrated for sound level (SPL/Leq) measurement, you typically see a number greater than 100. You can edit the numbers in this column directly, but more often, the calibration routine fills them in. For more information on this, see *Sound Level Calibration* on page 67.
- If the selected input device is a Smaart I-O, another way of calibrating for sound level measurement is to assign a named microphone in the *Microphone* column and let Smaart calculate the necessary calibration offset. If your microphone isn't already in the list but you know its sensitivity, you can create a named microphone by clicking the *Microphones* button below the channels table. In the

Microphones dialog, click the *Add* button, to name the microphone and enter its sensitivity in dB/Pascal. You should then see your new entry listed in the *Microphones* dialog and back in *I-O Config*, it will appear in the list when you click on any line in the *Microphone* column of the channels table.

- The *Mic Correction Curve* selectors assign microphone correction curves to input channels. For more information on importing microphone correction curves please refer to the topic on *Mic Correction Curves below*.

Figure 39: The Microphones dialog for managing named microphones

Below the channels table are several buttons whose functions are as follows:

- The *Clear Settings* button will clear out any calibration offsets, microphone and correction curve assignments that you have made and reset friendly names for channels to their driver-reported default names. (A warning message pops up first in case you click the button by accident.)
- The *Calibrate* button opens the *Amplitude Calibration* dialog with the currently selected input selected for calibration. For more information on calibrating input channels in Smaart, please see *Sound Level Calibration* on page 67.
- The *Mic Correction Curves* button opens the *Mic Correction Curves* dialog (see below).
- The *Microphones* button opens the *Microphones* dialog as discussed above.
- The *SPL Log Config* button opens the Log Config dialog where you can select the input channels for sound level logging (see *Sound Level Logging*, beginning on page 65, for more information).
- If the selected device is a Roland® OCTA-CAPTURE™, an additional check box labeled *Gain Tracking* appears. When this box is checked Smaart will adjust the calibration offset values for calibrated inputs on the device in response to gain changes. Note that the accuracy of these adjustments is dependent on the precision of the gain steps on the individual device, which can vary somewhat from one device to another. For more information, please refer to *Sound Level Calibration,* beginning on page 67.

Microphone Correction Curves

If you have individually measured frequency response data for your microphone, Smaart can use this information to flatten out any lumps and bumps in the microphone's response curve in spectrum and transfer function magnitude measurements. Microphone correction curves can be imported from comma- or tab-delimited ASCII text files having one frequency (in Hertz) and one magnitude value (in dB) per line.

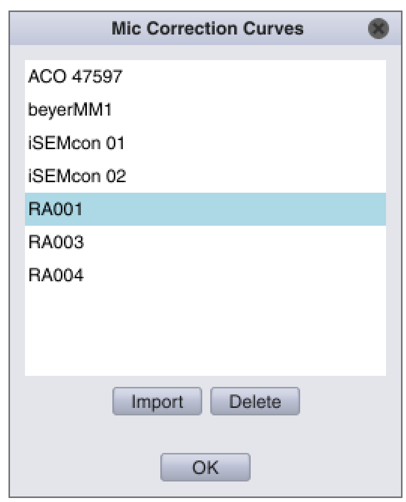


Figure 40: The Microphone Correction Curves dialog

You can import a new correction curve by selecting *Import* > *Mic Correction Curve* from the *File* menu or by clicking the *Import* button in the *Mic Correction Curves* dialog. Either action opens the *Import Mic Correction Curve* dialog where you can navigate to the file containing your correction curve and open it. If the import is successful, your curve should immediately show up in *Mic Correction Curves* dialog and in *Mic Correction Curve* lists in *I-O Config*. If not, the problem is likely just a formatting error of some kind. For more information on formatting correction curve files, please refer *Appendix G*, beginning on page 187. The *Mic Correction Curves* dialog is accessible by selecting *Mic Correction Curves* from the Options menu or by clicking the *Mic Correction Curves* button below the channels table on the *I-O Config* tab of the *Configurator*.

Measurement Config

The *Measurement Config* page of the *Configurator* dialog is central headquarters for configuring and managing live spectrum and transfer function measurements, along with tabs and windows to contain them. It is accessible by clicking the button on the *Control Bar* with the hammer and wrench icon, by using the keyboard shortcut [Alt/Option] + [G], or by selecting *Measurement Config* from the *Config* menu. You can also jump directly to the measurement settings for a specific measurement engine by double-clicking its control block on the *Control Bar*.

Tree Control

The *Measurement Config* page is divided into two main sections. On the left is a "tree" view of all windows, tabs and measurements that you have configured. The tree control can be used to create windows and tabs, and to copy or move measurements from one tab to another, or entire tabs from one window or another. The little plus or minus (+/-) boxes next to each tab name in the tree view are buttons that expand or collapse its contents. Double-clicking any window, tab, or measurement name in the tree view makes the name editable. As with most text fields in Smaart press the [Enter] to set your changes after editing a name.

The tree view also serves as a navigation bar for selecting what you see on the right side of the *Measurement Config* page, which you could think of as the "content" pane. Clicking on any tab or measurement name in the tree control displays the contents or settings for the selected item in the area on the right (when a window name is selected, the contents of its first tab are shown).

Two of the buttons below the tree view pane (*New Tab* and *New Window*) echo the functions of commands in the *Config* menu. As the names might imply, *New Window* creates a new Smaart window and *New Tab* creates a new empty tab in the selected window.

The *Copy* button below the tree view pane is a special case. You will note that it "latches" on or off when you click it. The state of this button determines the function of drag and drop mouse operations in the tree

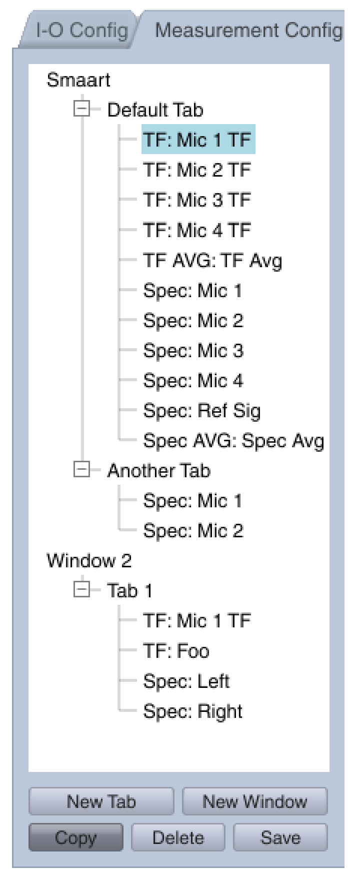


Figure 41: Tree view pane on the Measurement Config page of the Configurator dialog

view. When the button is engaged, as shown in the figure above, clicking and dragging any item in the tree view (a measurement, a tab or an entire window) from one place to another in the tree creates a new copy of the item when you release your mouse button to drop it. When not engaged, drag and drop operations *move* the selected item.

The *Delete* button deletes the selected window, tab or measurement. Please note that this action cannot be undone.

The *Save* button saves your entire Smaart setup, including all tabs, windows, measurements and display settings to a new named configuration that can be recalled later though the *Config Management* dialog, accessible from the *Config* menu (see *Managing Configurations* on page 46).

Tab View

When a tab name is selected in the tree control, you will see a table on the right side of the page like the one in the figure below, listing all of its measurements, their display colors, and the input channels driving them. Transfer function (*TF*) measurements appear in the top of the table with spectrum (*Spec*) measurements listed below. Double-clicking on a measurement in the table or selecting its name in the tree view will replace the measurements table with detailed settings for the selected measurement.

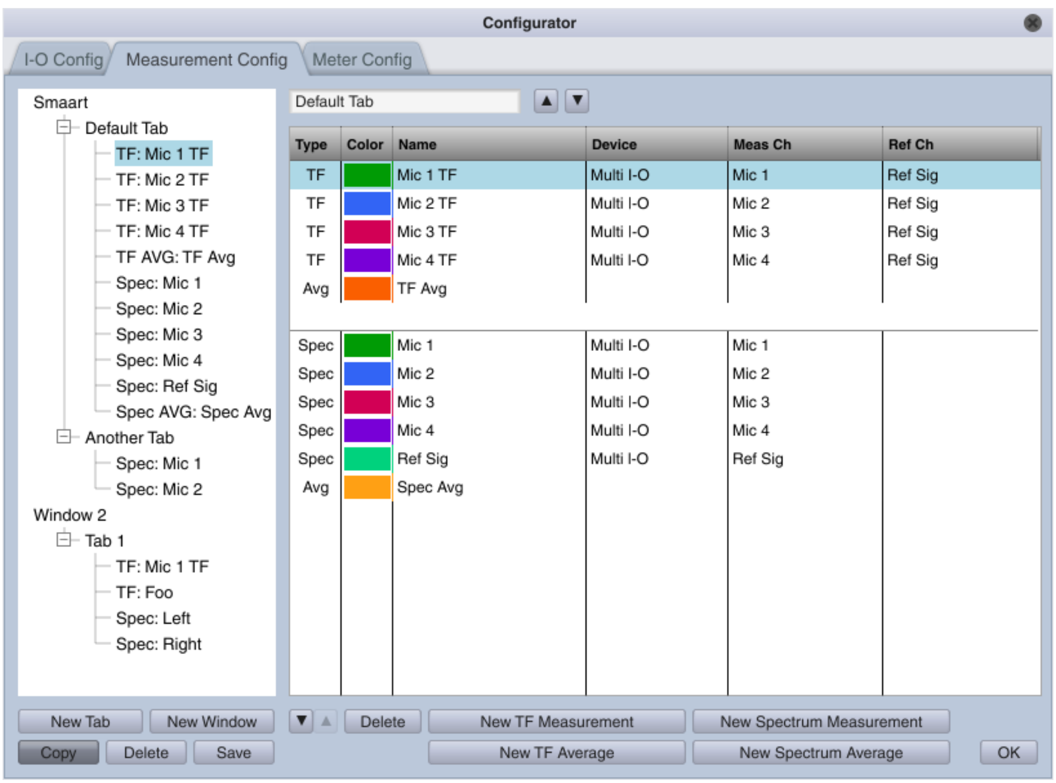


Figure 42: The Measurement Config page of the Configurator dialog

Below the measurements table are up/down (▲|▼) buttons for moving a selected measurement up or down in its list, a *Delete* button that deletes the selected measurement, and buttons for creating new spectrum and transfer function (TF) measurements and live averages. The latter have the same function as corresponding menu commands in the *Config* menu.

New TF Measurement opens the *New TF Measurement* dialog to create a new live transfer function measurement. For transfer function measurements, you just need to enter a name and select an input device and a pair of input channels to drive.

New Spectrum Measurement opens the *New Spectrum Measurement* dialog, where you can name your measurement and select the input device and channel to drive it.

New TF Average and *New Spectrum Average open* the *New Measurement Average* dialog where you can select other live spectrum or transfer function measurements to include in a real-time average. See *Live Averages* on page 42 for more information.

After creating a new measurement of any type, it will immediately appear in the measurements table above and in the tree view pane to the left. You can double click its name in the table or select its name in the tree view to see measurement settings in detail.

Spectrum and Transfer Function Measurement Settings

When a measurement name is selected in the tree view, settings for the measurement will appear on the right side of the *Measurement Config* page replacing the measurements table in the figure above. These include settings specific to each individual measurement and global settings that may apply to all measurements of the same basic type (spectrum or transfer function). Settings for both spectrum and transfer function measurements are divided into three groups of controls labeled *Measurement Settings*, *Input Settings* and *Global* (*Spectrum* or *TF)* *Settings*. The settings for spectrum and transfer function measurements are somewhat different. Spectrum measurements are the simpler of the two, so let's start there.

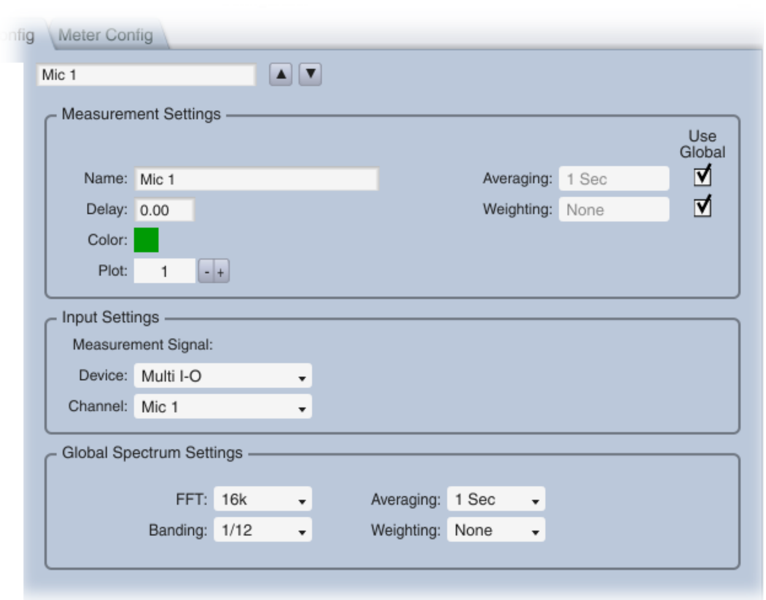


Figure 43: Detailed measurement settings for a spectrum measurement in Measurement Config

Spectrum Measurements

Measurement Settings

The *Name* field in the *Measurement Settings* control group sets the measurement name. If you edit the measurement name, be sure to press the [Enter] key to set the change when you are finished.

Note that when there are multiple copies of measurements with the same name in different tabs, changing the name of one instance of the measurement will unlink it from the others. Otherwise, the color and input settings for all instances are linked and changes to these settings affect all identically named copies. For auto-named spectrum measurements, created during the input selection process, changing the *Friendly Name* for the input channel driving the measurement(s) on the *I-O Config* page of the *Configurator* will automatically rename all copies of the measurement to match.

The *Delay* field sets the signal delay for the measurement in milliseconds. For spectrum measurements this will normally be 0.00, but there may be special cases you would want to delay a signal for display purposes. You can do that by entering a number in this field.

Clicking on the *Color* tile opens a color picker dialog to change the display color for a measurement. Changes to the display color will automatically be applied to all linked copies of the measurement with the same name.

The *Plot* control sets the preferred plot assignment for the measurement. It is ignored when only one chart of a given type is displayed. If you bring up a second chart of the same type in the same tab, measurements with a *Plot* setting of 1 (the default setting) will stay with the first instance of the chart and any measurements with a setting of 2 will move to the second.

Averaging specifies the length of time over which individual measurements are averaged to smooth and stabilize the traces or bar graphs on the screen.

Weighting applies a weighting curve to the spectrum measurement to reshape its spectrum, subtracting from the magnitude values at some frequencies and perhaps adding to them at others. Weighting in the frequency domain is analogous to filtering in the time domain. Common weighting curves include A and C weighting used for SPL and Leq measurements.

If the *Use Global* check box is checked for *Averaging* or *Weighting*, they will follow changes to the global settings for spectrum measurements (see below). If not, the measurement will keep its own setting(s) and ignore the global settings.

Input Settings

Input settings for spectrum measurements consist of a single input *Device* and *Channel* selection. As with *Color* selection (see above), changes to the settings in this section will be applied to all spectrum measurements with the same name.

Global Spectrum Settings

Two of the settings in the *Global Spectrum Settings* control group, *FFT* size and *Banding*, simply apply to all spectrum measurements in your configuration, *Averaging* and *Weighting* can be applied globally or they can be localized for individual measurements by un-checking their *Use Global* check boxes in the *Measurement Settings* control group (see above).

FFT size (in samples) determines the time constant and frequency resolution of the frequency domain displays. Increasing FFT size provides finer frequency resolution, enabling you to distinguish features more closely spaced in frequency, but does so at the expense of time resolution, the ability to resolve transient features in a signal that are closely spaced in time. In general, the default setting of 16K points is a pretty good trade-off between the two that works well for most applications, given a sampling rate of 44.1k or 48k samples per second.

Banding sets the frequency resolution for *RTA* and *Spectrograph* plots. Options include *None* (narrow-band frequency resolution), octave banding (*Oct*) and fractional octave banding from 1/3rd to 1/48th octave.

Transfer Function Measurements

Measurement Settings

The *Name* field in the *Measurement Settings* control group sets the measurement name. If you edit the measurement name, be sure to press the [Enter] key to set the change when you are finished. Note that when you have multiple copies of measurements with the same name on multiple tabs, changing the name of one instance of the measurement will unlink it from the others. Otherwise, the color and input settings for all instances are linked and changes to these settings affect all identically named copies.

Figure 44: Measurement settings for a transfer function measurement in Measurement Config

The *Delay* field sets the amount of signal delay (in milliseconds) needed to align the reference and measurement signals. Positive values delay the reference signal (the most common case). Entering a negative number delays the measurement signal.

Clicking on the *Color* tile opens a color picker dialog to change the display color for the measurement. Changes to the display color will automatically be applied to all linked copies of the measurement with the same name on other tabs and windows.

The *Plot* control sets the preferred plot assignment for the measurement. It is ignored when only one chart of a given type is displayed. If you bring up a second chart of the same type in the same tab, measurements with a *Plot* setting of 1 (the default setting) will stay with the first instance of the chart and any measurements with a setting of 2 will move to the second.

Checking the *Inverted* check box displays the measurement upside-down on magnitude graphs (only). This can be a handy option when setting loudspeaker EQ curves, as it may make it easier to match a hump in the response curve with a complimentary cut filter or perhaps set a boost filter to match a dip.

The settings on the right side of the transfer function *Measurement Settings* group can be assigned locally or globally. If the *Use Global* check box is checked for any of the following parameters, they will follow changes to the controls in *Global TF Settings* section (see below). If not, the measurement will keep its own settings and ignore the global settings.

FFT size (in samples) determines the time constant and frequency resolution of the frequency domain displays. When *MTW* is selected, Smaart performs multiple FFTs and combines the results into a single frequency data set. The individual FFT sizes used in this case are not user selectable.

Averaging specifies the length of time over which individual measurements are averaged to stabilize transfer function traces on the analyzer screen and improve their signal-to-noise ratio.

Phase Smoothing sets the amount of smoothing used for the transfer function *Phase* display. This option can be set globally or locally for each measurement.

Mag Smoothing sets the degree of smoothing type for transfer function magnitude traces. This option can be set globally or locally for each measurement.

Weighting applies a weighting curve to the measurement to reshape its spectrum/response curve, subtracting from the magnitude values at some frequencies and perhaps adding to them at others. Weighting in the frequency domain is analogous to filtering the measurement signal in the time domain. Common weighting curves include *A* and *C* weighting used for SPL and Leq measurements.

Mag Avg Type sets the type of averaging used for magnitude traces. The options are Polar (RMS) or complex (vector). Phase traces always use complex averaging. *Polar* averaging is probably the most common type for magnitude traces, but both types have their uses. In practical terms, *Polar* averaging lets more reverberant energy into the average, which may tend to agree better with what you hear, particularly for musical program material. *Complex* magnitude averaging tends to reject reverberant energy as noise and may give you better clues regarding speech intelligibility than polar averaging.

Input Settings

Input settings for transfer function measurements consist of *Device* and *Channel* assignments for the *Measurement Signal* and *Reference Signal*. The measurement signal will be the output of a device or system under test and the reference signal will be the input signal that produced that response. As with *Color* selection (see above), changes to the settings in this section will be applied to all transfer function measurements with the same name.

Normally, both signals will come from the same input device and so the *Device* selection for the *Measurement Signal* is automatically applied to the *Reference Signal* as well. It *is* possible to use signals from two different devices if you enable *Allow Multi-Device Transfer Function* in the *Advanced Signal Selection* section of *Transfer Function* options, however this will really only work if their sample clocks are synchronized. Even then, you may encounter issues with relative delay times changing when you stop and restart a measurement so proceed with caution if you decide to try this.

Global TF Settings

Two of the settings in the *Global TF Settings* control group, *Mag Threshold* and *Blanking Threshold*, apply to all transfer function measurements in your configuration. The rest can be all applied globally or localized for individual measurements by un-checking their *Use Global* check boxes in the *Measurement Settings* control group (see above).

Mag Threshold sets a normalized dB FS value that the reference signal must exceed before new data is accepted into the measurement at any given frequency. When the magnitude of the reference signal does not cross threshold at some frequency, new incoming data at that frequency is excluded from the average.

Blanking Threshold sets the coherence value that must be met or exceeded before a data point at a given frequency is displayed on the graph. This setting applies to both phase and magnitude graphs.

The remaining settings in the *Global TF Settings* control group apply to any transfer function measurement that subscribes to the global settings for *FFT*, *Averaging*, *Mag Avg Type*, *Phase Smoothing*, *Mag Smoothing*, or *Weighting*. These can also be set locally, at the individual measurement level and they are discussed in detail in *Measurement Settings* for transfer function measurements, beginning on page 57.

Live Average Settings

Measurement settings for spectrum and transfer function live averages are similar enough that we can talk about them together. Like other live measurements, they have a *Name*, *Color* and *Plot* preference. You can edit the *Name* field by clicking on it (press the [Enter] key when done to set the change) and clicking the *Color* tile pops up a color picker dialog where you can change the trace color.

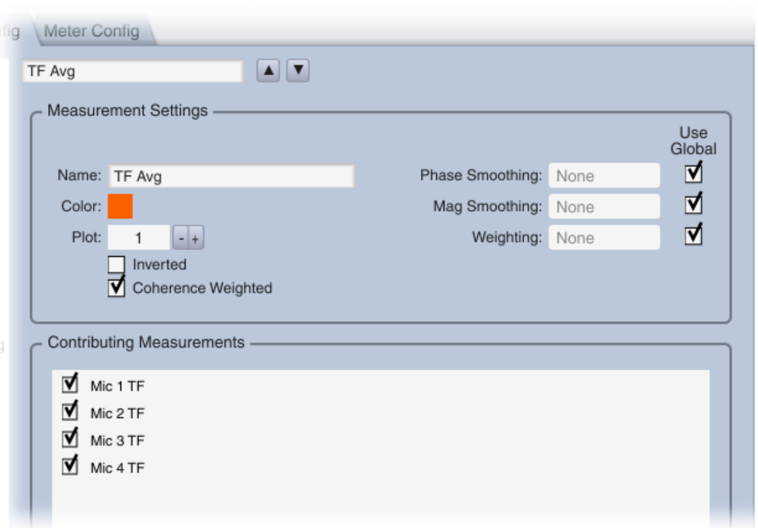


Figure 45: Detailed measurement settings for a live averaged measurement in Measurement Config

If you have copies of an averaged measurement with the same name on multiple tabs, changing the name of one instance of the measurement will unlink it from the others. Otherwise, the *Color* settings for all instances are linked and changing the color in in any copy affects all identically named copies. Note that if you copy a live average from one tab to another in the tree control Smaart automatically copies all of its contributing spectrum or transfer function measurements. You can only create live averages from measurements that reside on the same tab as the average.

The *Plot* control sets the preferred plot assignment for the measurement. It is ignored when only one chart of a given type is displayed. If you bring up a second chart of the same type in the same tab, measurements with a *Plot* setting of 1 (the default setting) will stay with the first instance of the chart and any measurements with a setting of 2 will move to the second.

Transfer function averages have an *Inverted* check box that turns the measurement upside-down on magnitude graphs (only) and also a *Coherence Weighted* check box. Coherence weighting gives more weight to the frequencies in each measurement that have the highest coherence values. As coherence tends to be a predictor of a signal-to-noise ratio in transfer function measurements, higher coherence suggests more trustworthy data. If a contributing measurement has poor coherence at some frequencies, for example, due to a localized reverberant buildup, coherence weighting will result in the more trustworthy frequencies contributing more to the overall average than its problematic frequencies.

Spectrum averages give you a choice of *Power* or *Decibel* averaging. *Power* averaging you the average power spectrum of the signals being analyzed and would be the typical choice for signal analysis applications such as background noise survey or for checking average sound level across a wide area for any other reason. Power averaging gives more weight to the loudest sounds in each frequency and may result in a graph that looks like it sounds.

Both spectrum and transfer function averages give you a choice of *Power* or decibel (*dB*) averaging. Power averaging gives you the average power spectrum of the signals being analyzed and would be the typical choice for signal analysis applications such as background noise survey or for checking average sound level across a wide area for any other reason. It tends to give more weight to the loudest sounds and may result in a graph that looks like it sounds. Decibel averaging is a simple arithmetic average of decibel magnitude values that gives equal weight to all magnitudes. You might say that it gives you more of a “consensus” view than power averaging.

For transfer function averages, you have the additional option of a *Coherence Weighted* decibel average. Coherence weighting gives more weight to the frequencies in each measurement that have the highest coherence values. Since coherence tends to be a predictor of a signal-to-noise ratio in transfer function measurements, higher coherence suggests that the data may be more trustworthy. Note that transfer function power averages are level adjusted (normalized) prior to averaging, based on the average decibel level of each individual trace in the range between 225 Hz and 8.8 kHz. Spectrum power averages omit this step because the levels are absolute in that case and level adjustment would not be desirable for all applications. For more information on spatial averaging options in Smaart, please refer to *Spatial Averaging,* beginning on page 19.

Both spectrum and transfer function averages have a *Weighting* control. Transfer function averages also have settings for *Mag Smoothing* and *Phase Smoothing*. can be set globally or locally for each measurement. If their *Use Global* check box is checked, they will follow changes to the global settings for measurements of the same type. If not, the measurement will keep its own settings and ignore the global settings.

Weighting applies a weighting curve to the measurement to reshape its response curve, subtracting from the magnitude values at some frequencies and perhaps adding to them at others. Weighting in the frequency domain is analogous to filtering the measurement signal in the time domain. Common weighting curves include *A* and *C* weighting used for SPL and Leq measurements.

Phase Smoothing sets the amount of smoothing used for the transfer function *Phase* display. *Mag Smoothing* sets the degree of smoothing type for transfer function magnitude traces.

SPL Config

The *SPL Config* page of the *Configurator* dialog is accessible by selecting *SPL Config* from the *Config* menu or by pressing [Ctrl/Cmd] + [Shift] + [E] on your keyboard. Settings on this page control the layout of the *SPL Meters* window, sound level logging functions, and the appearance and behavior of both in-tab SPL meter panes in main Smaart program windows as well as the individual meter modules in the *SPL Meters* window. The *SPL Config* tab is organized into six sections.

SPL Display Settings

The upper left section of *SPL Config*, labeled *SPL Display Settings*, controls the overall layout of the SPL Meters window and display colors for the meter readings. To configure the SPL meters window, you begin by setting both of the *Display Grid* fields to a number greater than zero. Together, they determine how many meters the window will contain and how they are laid out. The first number sets the number of columns and the second is the number of rows. The total number of meters in the *SPL Meters* window will be the first number multiplied by the second. In the example below, we have set up a meter window that is two modules wide by two rows deep (2 x 2), for a total of 4 meter modules.

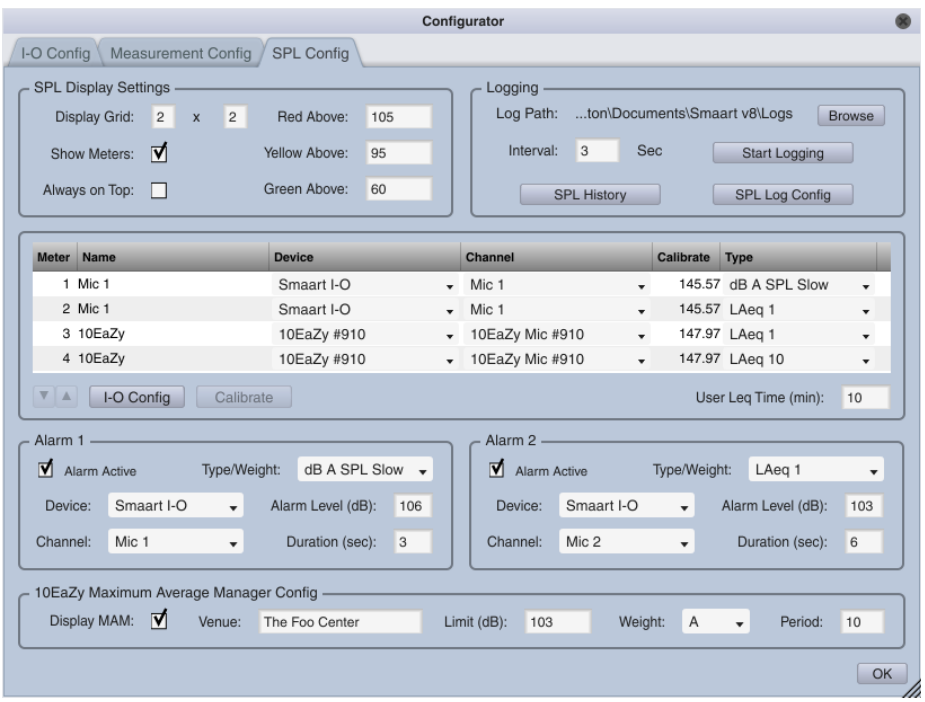


Figure 46: SPL Config page in the Configurator dialog

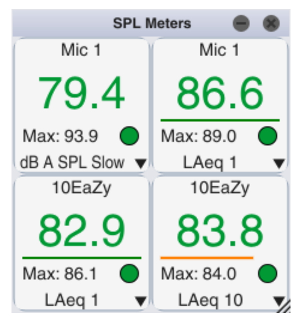


Figure 47: 2x2 SPL Meters panel with SPL and LEQ meters shown

Show Meters opens the *SPL Meters* window when checked and closes it when un-checked. Note that you can also do this from the main Smaart window by pressing the [E] key on your keyboard or selecting *SPL Meters* from the *View* menu.

Always on Top makes the SPL Meters window stay in front of all the other windows on your screen when checked (except the Configurator window, ironically), regardless of which window has focus.

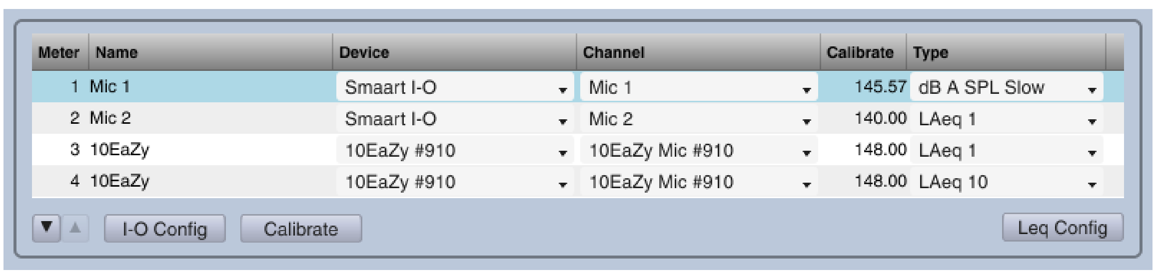
The *Red Above*, *Yellow Above* and *Green Above* settings set sound level thresholds for SPL and Leq meters where the readout text will change color. Typical usage for this feature would be to set the *Green Above* level comfortably above the self-noise level of your microphone and preamp, so that when the meter level is green, you will know that you are measuring actual sounds. The *Yellow Above* threshold might then be set to a sound level that should not be exceeded and *Red Above*, to a level that must not be exceeded. Note that the SG Audio Aps 10EaZy reads its usable range figure directly from the device and ignores the global *Green Above* threshold. Otherwise, these settings apply to all SPL and Leq meters in Smaart, including the in-tab SPL meter panes in main program windows.

Logging

Controls in the upper right section of *SPL Config* provide access to Smaart's sound level logging functions. Logging functions are covered in detail later in this chapter. Please refer to *Sound Level Logging*, beginning on page 65, for more information.

Meters Table

In the center of the *SPL Config* page is a table listing all of your meters and their individual settings. Each line in the table defines a meter module for the *SPL Meters* window. Most of the entries in this table, with the exception of the *Meter* number and *Calibrate* columns, are interactive controls. The number of lines in the table is determined by the *Display Grid* settings in the *SPL Display Settings*.



Meter	Name	Device	Channel	Calibrate	Type
1	Mic 1	Smaart I-O	Mic 1	145.57	dB A SPL Slow
2	Mic 2	Smaart I-O	Mic 2	140.00	LAeq 1
3	10EaZy	10EaZy #910	10EaZy Mic #910	148.00	LAeq 1
4	10EaZy	10EaZy #910	10EaZy Mic #910	148.00	LAeq 10

Figure 48: Detail of the meters table in SPL Config

When you create a new set of meters, Smaart picks a default input device and channel and picks up the *Friendly Name* assigned to the input channel in *I-O Config* (see *Configuring Input and Output Channels* on page 50 for details) as the meter name. As long as a meter name is identical to its input channel name, any changes to the input channel name will flow through to the meter. This, however, is not a requirement. Meter names can be anything you like. Just click on any entry in the *Name* column to change it – as with most text fields in Smaart, remember to press the [Enter] key after editing to apply the change. The meter will then ignore any subsequent changes to the input name unless you change either one to match the other, which will re-link them.

Clicking on any entry in the *Device* column pops up a list of all configured input devices wherein you can select the input device that you want to use for the meter module. Likewise, each entry in the *Channel* column is a drop list control that selects which input channel on the selected device to use. Input devices and channels are listed by their friendly names and only the devices and channels selected for use on the *I-O Config* page of the *Configurator* will appear in these lists, so if you don't see the one that you are looking for, switch to the *I-O Config* tab and make sure its *Use* box is checked. Note that when a meter's *Name* (see above) is identical to its input channel name, changing the input channel assignment will also change the meter name to match the new selection. If not, then the meter name is unaffected.

Entries in the *Calibrate* column show the offset in decibels from digital full scale (FS) assigned to calibrate each meter's input channel for sound level measurement (if applicable). These are directly editable except in the case of the SGAudio Aps 10EaZy or a Rational Acoustics Smaart I-O input channel that has been calibrated by microphone sensitivity. Microphone assignments for the Smaart I-O can be changed from the *I-O Config* page of the *Configurator* or the *Amplitude Calibration* dialog. Calibration offset for the 10EaZy can only be changed by recalibrating.

When the *Calibrate* adjustment for an input channel is set to 0 dB, it means that the input is calibrated internally to full scale and the only measurement type available for the meter will be dB FS (peak). To calibrate an input for sound level measurements, you can click on its *Meter* number to select the row and then click the *Calibrate* button below the table to open the *Amplitude Calibration* dialog. For details on calibrating Smaart for sound level measurements, please refer to the section on *Sound Level Calibration*, beginning on page 67.

The drop list controls in the Type column set the measurement type for each meter. The choices are *FS Peak* for a simple peak-reading signal level meter referenced to normalized digital full scale or, if the input is calibrated for sound level measurement, any of 11 built-in sound level measurement types plus any user-defined Leq measurements that you may have set up. Built-in types include A-weighted, C-weighted or unweighted SPL (with standard Fast or Slow exponential time integration), Leq with a one-minute (Leq 1) integration periods, and Peak C.

To add user-defined Leq options, click the *Leq Config* button, located below the meters table on the right, to open the *Leq Config* dialog. In the dialog window, click the plus (+) button to add a new list entry and then edit the *Weight* and *Time* fields for the new entry by clicking them with your mouse. After setting your desired integration time, be sure to press the [Enter] key on your keyboard to set the change. To delete an existing entry, select it in the list by clicking with your mouse and then click the minus (−) button. When you click the OK button to exit the dialog window, the new Leq type(s) that you defined will be immediately available in applicable selector menus.

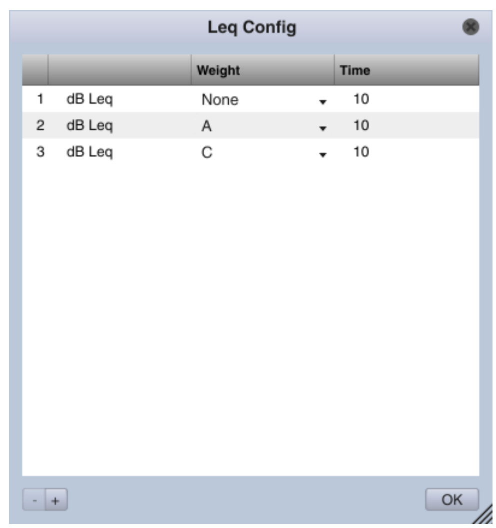


Figure 49: Leq Config dialog with three custom Leq types defined (Leq 10, LAeq 10, and LCeq 10)

Note that you can also select measurement type for SPL meters on the fly, directly from the *SPL Meters* window, using the drop list control integrated into each meter module (see *Figure 30* on page 41). Smaart continuously calculates all of the available measurement types in the background regardless of which type is being displayed, meaning that you can see current results immediately when switching between types, without having to wait for the average to repopulate each time.

Immediately below the meters table on the left is a pair of up/down (▲ | ▼) buttons that move a selected meter definition up or down in the table. This reorders the position of the corresponding meter in the *SPL Meters* window as well. To the right of those buttons are two more, labeled *I-O Config* and *Calibrate*. These provide shortcuts to input and calibration settings for the selected meter (see *Audio I-O Configuration*, beginning on page 50, and *Sound Level Calibration* on page 67 for more information).

The *Leq Config* button opens the *Leq Config* dialog (see above) for defining custom Leq measurement periods.

Alarms 1 and 2

Below the table of meter configurations are two control groups labeled *Alarm 1* and *Alarm 2*. These can be used to set up thresholds for individual input channels that will trigger an alarm. When triggered, the alarm will cause the border of the SPL Meters window to flash red and alarms are flagged as discrete events in sound level log files when logging is turned on.

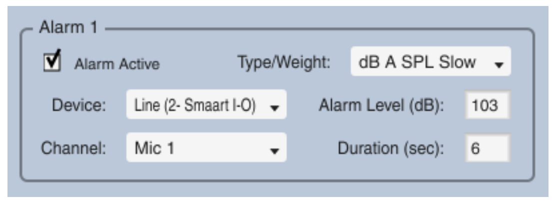


Figure 50: Alarm settings in SPL Config

To set an alarm, click its *Alarm Active* check box if it is un-checked, then use the drop list controls to select the input *Device* and *Channel* that you want to monitor and the measurement metric (*Type/Weight*) that you wish to use. Enter your desired threshold level in the *Alarm Level* field and press the [Enter] key to apply the setting. The *Duration* period sets the number of seconds that the alarm will continue flashing after the sound level at the input being monitored drops back below the alarm threshold.

10EaZy Maximum Average Manager Config

The settings in the bottom section of *SPL Config* are active only when one or more SGAudio Aps 10EaZy devices are connected to your computer and selected for use on the *I-O Config* tab of the *Configurator*. When active, these settings apply to all available 10EaZy devices, if you have more than one.

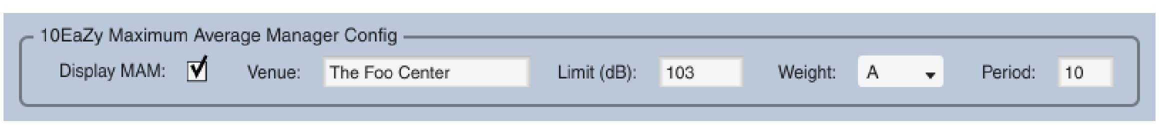


Figure 51: 10EaZy MAM controls in SPL Config

The 10EaZy Maximum Average Manager (MAM) is a predictive algorithm designed to help live sound engineers keep the output of a sound system within a prescribed sound level limit, specified in terms of equivalent sound level (Leq) such as Leq 10 or Leq 15. Setup is simple, requiring only 3 pieces of information: The equivalent sound level (Leq) *Limit* in decibels that you wish to stay below, the Leq *Period* in minutes and the frequency weighting curve (*A*, *C* or *None*) that you want to use for the Leq. For more information on using the 10EaZy MAM display in Smaart, please refer to 10EaZy Maximum Average Manager (MAM) in chapter two, beginning on page 42.

Sound Level Logging

Figure 52: Log Config dialog

Smaart can perform sound level (SPL and Leq) and full scale signal level logging on any calibrated input that is selected for use in *I-O Config*. "Calibrated," in this context, simply means that the calibration offset (Cal. Offset) specified for the input in *I-O Config* is greater than 0 dB. If the *Cal Offset* figure for a given input channel is set to something other than zero, Smaart makes it available for logging. Of course, sound level measurements are only *accurate* if the input is calibrated accurately (*see Sound Level Calibration* on page 67 for details) and Smaart has no way of knowing if that is true, so it is the operator's responsibility to ensure that the calibration figures are correct for their application.

Having confirmed that your calibration settings are in order, the first step in setting up sound level logging is to select the input(s) that you want to log in the *Log Config* dialog, which is accessible from the either *I-O Config* or *SPL Config* tab of the *Configurator* dialog. To open the *Log Config* dialog from the *I-O Config* tab, click the *SPL Log Config* button below the channels table. On the SPL Config tab, the button to open *Log Config* is located in the *Logging* control group, in the upper right corner of the page.

To select an input channel for logging, simply click the check box next to its name in the *Log Config* dialog. Or, you can select all calibrated inputs on a given input device by clicking the check box next to the device name. If you like, you can also fill in the three text fields in the upper part of the dialog window. Your entries here will then appear in all log file headers. Note that input devices and channels are listed by their friendly names. If you don't see a device or channel that you want to log, check its *Cal. Offset* and *Friendly Name* settings and make sure that its *Use* check box is checked in *I-O Config* (see *Audio I-O Configuration*, beginning on page 50, for details). When you have made your selections in Log Config, click the OK button to apply your changes and exit the dialog, then switch to the *SPL Config* tab of the *Configurator* dialog (if you are not there already).

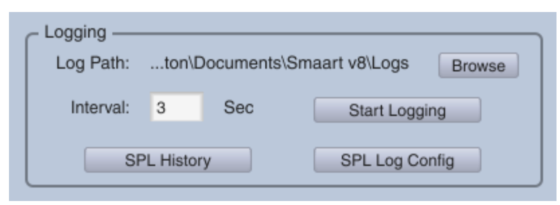


Figure 53: Logging controls in SPL Config

Logging controls in *SPL Config* are located in the aptly named *Logging* control group on the upper right. The *Log Path* shows the name of the folder currently selected for sound level log files. The default location is a folder named "Logs" in the Smaart v8 folder, which is located in the default Documents directory for your user account. If you prefer to store your log files elsewhere, you can change the current *Log Path* by clicking the *Browse* button and then navigate to the folder where you want your log files to reside.

The *Interval* setting controls how often log files are updated. The default is every 3 seconds.

Logging begins when you click the *Start Logging* button and continues until you explicitly turn it off or exit the program. If you exit the program with logging turned on, logging will resume automatically when the program is restarted, making it possible to set up scheduled operations or auto-recover from system crashes or power failures using your OS scheduler or other third-party tools.

The *SPL History* button opens the graphical *SPL History* display window, which was covered in chapter two (see *SPL History Window*, beginning on page 43, for more information on this display). The *SPL Log Config* button opens the *Log Config* dialog as discussed earlier.

Log File Format

Smaart sound level log files are written in plain, tab-delimited ASCII text format. When logging is turned on, Smaart creates a separate file in the designated Log Path folder (see above) for each input channel that has been selected for logging.

Completed log files consist of a header block followed by a data table containing 14 types of directly measured data recorded at each logging interval. These include A and C-weighted and unweighted sound pressure levels with standard Fast and Slow exponential time integration, A and C-weighted and unweighted Leq for each of two integration periods (1 minute and user-specified), C-weighted peak sound level (Peak C) and unweighted, full scale peak signal level. For the SPL levels, Smaart uses the maximum reading recorded during each logging interval. Leq levels are simply the Leq at the time each line in the file was written.

In addition to the SPL and Leq data, C-A levels for each Leq period are calculated for each logging interval and four additional columns record alarm events, input overloads and Leq buffer resets. C-A levels are simply the difference between A-weighted Leq and C-weighted Leq, which provides a rough estimate of the low-frequency content of the sounds being measured. Alarms, overloads, and Leq resets are marked by asterisks.

Log files are named automatically using the date and time (year, month, day, hour, minute), followed by the input device and channel friendly names assigned in I-O Config.

Log data files: **YYYYMMDD.HHMM.DeviceName.InputName.data.txt**

Completed log files: **YYYYMMDD.HHMM.DeviceName.InputName.txt**

While logging is in progress, the log data file will contain only the data table. When you turn logging off or exit Smaart normally, the header blocks are written and the completed files are renamed automatically. In the event of a program or system crash or power loss, header blocks will not be written but all measurement data collected can still be found in the log data file.

Header files are comprised of environment information, max levels and event counts. Environment data includes the file creation date, the operator, company and venue names from the *Log Config* dialog, input device and channel names, the calibration date if known, and the Smaart software version number. Calibration dates for each input channel are based on the last time the calibration offset for the channel was changed and are stored in the Smaart config file. The max levels and event summaries section includes the *Alarm 1* and *Alarm 2* levels specified in *SPL Config*, the total number of times each alarm was triggered and the total number of times the input channel was overloaded along with max levels recorded for the entire log for each SPL and Leq type.

If the input device being logged is an SGAudio Aps 10EaZy, an additional block of information will appear in the header with settings from the *10EaZy Maximum Average Manager Config* section in *SPL Config*, the calibration type (factory or user) and the number times the specified MAM limit level was exceeded.

Sound Level Calibration

To Calibrate one or more input channels for sound level (SPL or Leq) measurement in Smaart, press [Alt] + [A] on your keyboard or select *I-O Config* from the *Config* menu to open the *Configurator* dialog to the *I-O Config* page. In I-O Config, select the input device and channel that you want to calibrate and click the *Calibrate* button at the bottom of the page below the channels table. This opens the *Amplitude Calibration* dialog.

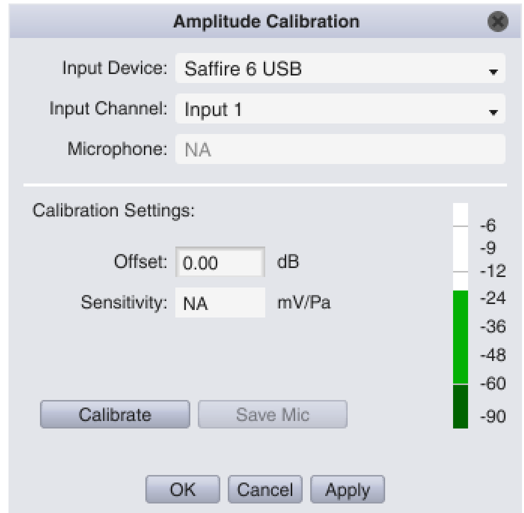


Figure 54: The Amplitude Calibration dialog

At the top of the calibration dialog are three drop-list selectors for *Input Device*, *Input Channel* and *Microphone.* Make sure that the device and channel that you want to calibrate are the selected – if not you can change the selections. If the input device that you are working on happens to be a Smaart I-O, then the *Microphone* selector becomes available as well, otherwise it will be disabled.

Calibrating with a Sound Level Calibrator

In the realm of digital audio, the only real reference we have for amplitude values is how big they are relative to the biggest number that you can get from a sample word of a given number of bits. For example, a 24-bit signed integer sample word gets you a maximum PCM amplitude of +/- 8388607 (2^{23} - 1). We typically normalize those maximum full-scale numbers to +/- 1, so that full scale works out to be 0 dB and all lesser amplitudes are negative numbers on a decibel scale. In order to relate that internal amplitude reference to some number of volts or Pascals of pressure in the real world, we need to calibrate it to signal of known amplitude – and to do *that* for a microphone we need an *acoustical* signal of known amplitude. This is where a sound level calibrator comes in.

Calibrating an input channel and microphone for SPL measurement using a sound level calibrator is a two-step process consisting of:

- Measuring the Full Scale signal level of a microphone with the sound level calibrator coupled to it
- Assigning the reference amplitude value for the sound level calibrator to the measured Full Scale amplitude

To measure the input signal level, you need to:

1. Connect a microphone to the input channel that you want to calibrate.
2. Affix your sound level calibrator to the microphone.
3. Turn on the calibrator and adjust the gain for your input channel to a desirable level.
4. Click the *Calibrate* button in the *Amplitude Calibration* dialog to run Smaart's calibration routine.

Bear in mind that once you are calibrated, you will need to leave the input gain set exactly where it is to maintain calibration (unless you are using a Smaart I-O) and so a little forethought with regard to setting levels might save you needing to repeat this procedure again later. Two things to consider are the loudest sounds that you need to measure and the maximum SPL rating for your microphone.

If your microphone were rated for 120 dB max SPL, then you would properly need a different microphone to measure sounds louder than that. On the other hand, if your microphone is rated for 140 dB, then hopefully, you won't need to get anywhere near that and you might want to pick a lower number as your target maximum. Whatever you decide on as your max SPL figure, subtract it from the reference level for your calibrator (e.g., 94, 104 or 114 dB) and if the result is a negative number, then that is your maximum *full-scale* amplitude for calibration. If your target full-scale amplitude works out to 0, then 0 dB full scale would be your absolute max, but you might want to call it something more like -1 or -2 dB FS, just to make sure you don't clip the input during calibration.

The *Amplitude Calibration* dialog has an input level meter that shows you the peak full-scale signal level for the selected channel (see ***Error! Reference source not found.***). With your microphone calibrator running, adjust the gain for the input channel to your target full-scale amplitude level and then click the *Calibrate* button. The *Calibration Progress* dialog pops up, Smaart measures the input signal over a period of a few seconds, and then reports the full-scale signal level. If you are happy with the result, make sure the value in the *Set this value to* field in the pop-up dialog matches the reference level of your calibrator, and then click *OK*.

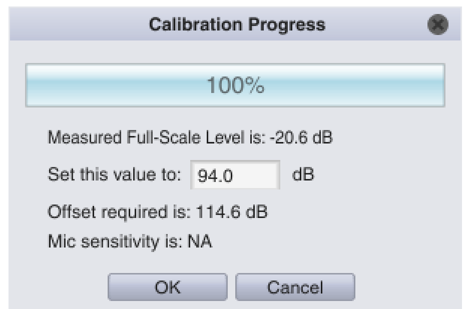


Figure 55: The sound level Calibration Progress dialog

Back in the *Amplitude Calibration* dialog, you will see that Smaart has calculated the necessary *Offset* value to calibrate the selected input to SPL. If the input device is a Smaart I-O, the *Sensitivity* field will be populated as well. If you are using a Smaart I-O, you can click the *Save Mic* button and give it a name and Smaart will select the new microphone name automatically. Otherwise, we are finished. You can click *OK* to exit the dialog or click the *Apply* button and select another input to calibrate.

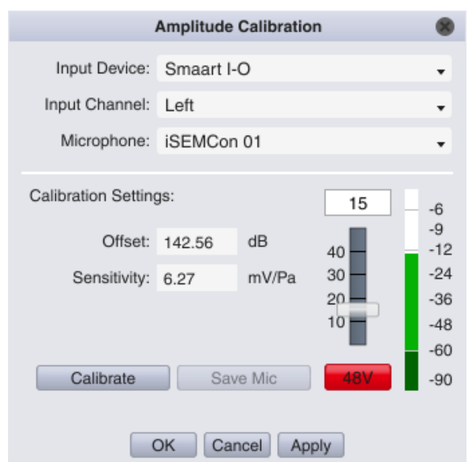


Figure 56: Amplitude Calibration dialog with a gain and phantom power controls for a Smaart I-O input channel.

Calibrating Based on Microphone Sensitivity (Smaart I-O Users)

The Smaart I-O is a special case for calibration because Smaart knows the electrical sensitivity of its inputs and can read its preamp gains settings. This makes it possible to calculate the combined sensitivity of the preamp and microphone, provided that the microphone sensitivity is known. When the selected input device in the *Amplitude Calibration* dialog is a Smaart I-O, a gain control and 48 V phantom power button appear beside the input level meter and the *Microphone* selector and *Sensitivity* fields become enabled.

Measurement microphones often come with individually measured sensitivity and frequency response data, so if you know the sensitivity of your microphone in millivolts per Pascal, you can just enter the number in the *Sensitivity* field and press the [Enter] key on your keyboard to set the change. Smaart will calculate the required offset for SPL calibration. If you do not know your microphone's sensitivity, you can follow the procedure for *Calibrating with a Sound Level Calibrator* (above) to measure it. Either way, once you have the sensitivity value filled in, you can click the *Save Mic* button, to give it a name and save it to your microphones list. You will then be able to calibrate the input of the Smaart I-O for that Mic in the future by selecting its name from the *Microphone* list in *I-O Config* or in the *Amplitude Calibration* dialog.

Notes on Gain Tracking for the Roland® OCTA-CAPTURE™

When the selected input device is a Roland OCTA-CAPTURE Smaart will detect changes and automatically adjust the calibration settings for calibrated input channels to compensate, provided that the *Gain Tracking* check box for the device, located under the input channels table for the device in I-O Config, is checked. The accuracy of these adjustments is device-dependent but will typically range within a maximum error of about +/− 1.5 dB from nominal gain setting, between 0 and 50 dB, when calibrated at a gain setting of +25 dB.

To quickly determine the maximum gain adjustment error for your specific device, set input to a gain to 25 dB, then calibrate for sound level measurements using a sound level calibrator (see page 67) with *Gain Tracking* turned off. After calibrating, with the calibrator still fitted to the microphone and turned on, enable *Gain Tracking* and watch the SPL meter in Smaart as you turn the gain up to 50, and down to 0 dB. The maximum deviation observed from the reference level of the calibrator represents the maximum expected accuracy of your setup.

Chapter 4: Real-Time Mode User Interface

Real-Time Mode Main Window Layout

The first time you run Smaart, you will find yourself looking at a screen similar to the one below, except that we have taken a few liberties here for presentation purposes. We have switched from the Default Dark color scheme to the higher contrast Default Light scheme (*View* menu > *Skins* > *Default Light*), we have two live measurements set up and running, and there is captured data in the Data Bar.

Figure 57: Anatomy of the main window layout for real-time mode

1 Tab Bar

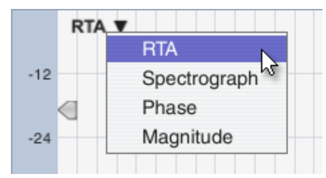
Smaart can run in multiple windows and each window can host multiple tabbed workspaces that we refer to simply as tabs. Each tab includes its own measurements, screen layout, graph assignments and data file show/hide selections. You can switch between tabs by clicking the tab-shaped buttons below the menu bar in the area we call the Tab Bar. You can move a tab from one Smaart window to another by clicking on its button in the Tab Bar with your mouse and dragging it to another window, then releasing the mouse button to drop it.

If you are not using multiple tabs in a window or are not switching between tabs very often, the Tab Bar can be hidden to make a little more room for graphs by selecting Tab Bar from the View menu or pressing the [A] key on your keyboard. Repeating either of these actions will restore the Tab Bar when it is hidden. Note that when the Tab Bar is hidden, you can still switch between tabs using the *Tab* selector on the Control Bar.

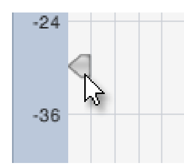
❷ Cursor Readout

When measurement data is present on a graph, the cursor readout displays numeric coordinates corresponding to the cursor location(s) as you move your mouse over the graphs areas. Numeric coordinates are provided here for the cursor location in units of amplitude/magnitude and frequency or time, as applicable to graph type. The cursor readout works much the same for all graph types in Smaart and it is covered in detail in the *Common User Interface Elements* section of Chapter 2, on page 28.

❸ Main Graph Area



The main graph area in real-time mode can be divided into one or two main graph panes (plus an optional Live IR pane for transfer function displays) using the display control buttons at the bottom of the Control Bar or by recalling a *View Preset* in the *View* menu. Main graph panes can be assigned any of the four real-time frequency chart types (RTA, Spectrograph, Transfer Function Magnitude or Phase) using the drop-down menu that appears in the upper left corner of each graph pane. For the Live IR pane you have your choice of three time domain graph types; impulse response with linear or logarithmic amplitude scaling or Envelope Time Curve (ETC).



The two arrowhead-shaped widgets that you can see positioned on the left edge of RTA and Spectrograph charts are threshold controls for the Spectrograph. You can click on these and drag them up and down with your mouse to set the minimum and maximum thresholds for the spectrograph dynamic range. A similar widget that appears on the right edge of transfer function Magnitude graphs is used to set the coherence blanking threshold.

When one or more live measurements or stored data traces are present on a graph, the name of frontmost trace in the *z*-axis stacking order (we also call this the top trace) appears in the upper right corner of the graph pane. You can also cycle the *z* order of a graph by pressing the [Z] key or use [Shift] + [Z] to cycle in the other direction. If there is a weighting curve applied to the measurement, the weighing curve name appears below the measurement name.

Graph Legends

Clicking on the name of the front trace opens the legend box for the graph, which lists all live measure-ments and captured traces that are currently visible on the graph. Clicking anywhere in the Smaart window outside the legend box closes it. Live measurement traces appear in the legend as round

buttons while captured traces are represented by file icons (a page with the corner folded over). One trace is always selected – normally the one at the top of the list – and the current *active live measurement* is indicated by an outline around it.

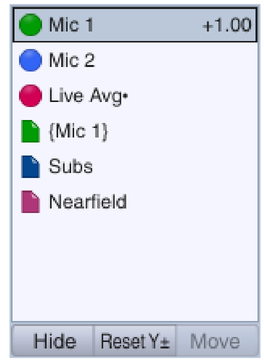


Figure 58: Legend box for a Transfer Function graph

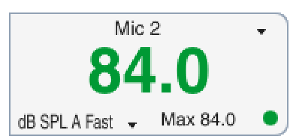
Each icon in the legend is colored to match the display color of the corresponding data trace on the graph. If any trace has a weighting curve applied to it, a small bullet (•) is appended to the trace name to indicate this. When a trace is displayed with vertical (*y*-axis) offset, the offset value in decibels is shown to the right of the name. An inverted transfer function trace is shown with its name enclosed in curly brackets (e.g., "{Mic 1}" in Figure 58).

You can hide a measurement by clicking its icon. When you do this, the trace is removed from the graph and an "X" is drawn though its icon on the control bar if it is a live measurement, or in the data library pane of the Data Bar if it is a captured data file. Clicking on the icon for a hidden data file or live measurement restores it to visibility on the graph.

Clicking on the *name* of a live or captured trace in the legend box selects the object and moves it to the top of the legend list. This also makes it the top trace in the *z*-axis stack on the corresponding graph(s). You can multi-select traces in the by holding down the [Ctrl/Cmd] key while clicking their names with your mouse, or hold down the [Shift] key while clicking to select a contiguous group of objects.

Below the legend list are three buttons. The *Hide* button hides a selected trace or group of traces and removes them from the graph. The *Reset Y±* button clears all vertical offsets applied to any live or captured data trace. This action can be undone while the legend remains open by clicking the *Reset Y±* button again, provided that no additional changes have been made to trace offsets in the meantime. When two graphs of the same type are selected, the *Move* button moves a selected trace or measurement from the current graph to the other.

❹ SPL Meter (or Clock)



The large numeric readout that appears (by default) at the top of the Control Bar in the upper right corner of the each tab can be configured to function as a Sound Pressure Level (SPL) meter, an integrating Equivalent Sound Level (Leq) meter, a peak signal level meter calibrated to normalized digital full scale, or a clock. When the level meter is displayed, pressing the [K] key on your keyboard switches the display to a clock and vice versa. This display can be hidden if you don't need it by selecting *SPL Meter* from the *View* menu pressing [Alt/Option] + [K] on your keyboard. When hidden, repeating either of these actions will restore it.

The in-tab SPL Meter operates almost identically to the meter module in the *SPL Meters* window. Both are covered in detail in the section on *Sound Level Metering* on page 40. Note that in order to perform accurate SPL or Leq measurements, the input being monitored must be calibrated to SPL. Please see *Sound Level Calibration* on page 67 for more information.

❺ Control Bar

The Control Bar in real-time mode is home to live measurement controls for the active graph, signal generator, and main display controls for real-time frequency-domain measurements and the Live IR. The Control Bar and in-Tab SPL Meter (when present) can be hidden by clicking the triangular button in the border between the Control Bar and the graph area. This button remains visible in the window border when the Control Bar is hidden and clicking the button again will restore it. You can also hide or restore the Control Bar by means of the *Control Bar* command in the *View* menu or by pressing the [O] key on your keyboard.

Live Measurement Controls

The Control Bar proper consists of live measurement controls for the active graph (see *Active Graph Pane* on page 29). When the active graph is an RTA or Spectrograph, this area contains controls for spectrum measurements. If the active graph is a transfer function Magnitude, Phase or Live IR graph, then transfer function measurement controls appear here.

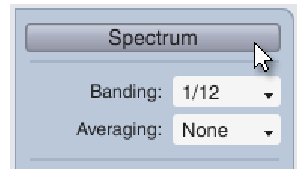


Figure 59: Active measurement controls for a spectrum measurement

The label at the top of this section indicates which type of display (*Spectrum* or *Transfer Function*) is currently active. Notice that the label becomes a button when the mouse cursor hovers over it. Clicking this button opens *Spectrum* options or *Transfer Function* options, depending on the current active graph selection.

The first group of controls applies to the active measurement, which is to say, the front-most live measurement on the active graph. The current active live measurement selection is indicated by the background color of its control block in the lower portion of the control strip. In the example shown here, measurement labeled *"Mic 1 TF"* is active.

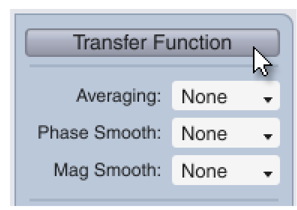


Figure 60: Active measurement controls for a transfer function measurement.

- The active measurement controls for a spectrum measurement consist of *Banding* and *Averaging* selectors.
- For transfer function measurements, the control set includes an *Averaging* selector and two separate smoothing controls for phase (*Phase Smooth*) and magnitude data (*Mag Smooth*).

If the current active measurement uses the global settings for averaging, banding or smoothing (as applicable), then changes to these settings will affect all measurements of the same type that also use the global settings. If the current active measurement is *not* subscribed to the global selections for a given setting, then the selector affects only the active measurement. For more information about these parameters, please refer to the *Measurement Settings* for spectrum and transfer function measurements in *Measurement Config*.

Note that when the *Enable FTW* check box is checked in *Transfer Function* options, two additional controls appear below the averaging and smoothing controls for transfer function measurements; a check box to turn FTW on and a text field to set the nominal time window size in milliseconds.

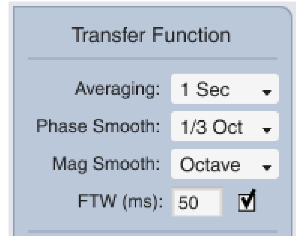


Figure 61: Transfer function active measurement control group with FTW smoothing controls.

Frequency-domain "time windowing" (FTW) is a complex linear smoothing technique performed in the frequency domain that is mathematically equivalent to applying a tapered window function to the impulse response in the time domain and transforming the result with a zero-padded FFT.

FTW is a global function applied to all live and captured transfer function measurements that use *Complex* magnitude averaging (only). Enabling FTW forces the global magnitude averaging selection for live transfer function measurements to *Complex*, however, live measurements that do not use the global setting are unaffected by FTW if their magnitude averaging selection is set to *Polar*. Stored data traces captured using polar magnitude averaging are also unaffected.

Note that when FTW is turned on, a vertical red line appears on transfer function Magnitude and Phase graphs denoting the low-frequency cutoff for the equivalent time window. Because FTW limits the effective window size of the equivalent impulse response, data at frequencies below the cutoff frequency cannot be reliably resolved and should not be considered trustworthy. For more information on FTW, see *Frequency-Domain "Time-windowing" (FTW), or Linear Complex* Smoothing on page 16.

Below the active measurement controls are tab-level measurement controls and individual control blocks for live measurement engines. The *Tab* selector can be used to switch between tabs if the *Tab Bar* is hidden. The button to its right labeled with the hammer and wrench icon opens the *Measurement Config* page in the *Configurator* dialog.

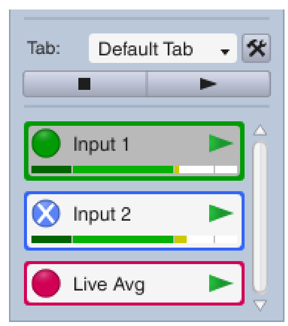


Figure 62: Live measurement controls for two spectrum measurements (above) and one live average.

Both spectrum and transfer function measurements have *stop all* (■) and *run all* (▶) buttons below the *Tab* selector that turn *all* measurements in the tab off or on. Tab-level transfer function measurement controls also include *All Track* and *No Track* buttons that turn delay tracking on and off for every measurement in the tab (see *Toggle Delay Tracking* for more information).

The lower portion of the live measurement controls section is a scrollable area containing the control blocks for each individual measurement in the tab. If you have more measurements configured than can fit here, a scroll bar appears so that you can scroll through the list.

The control blocks differ somewhat depending on the measurement type. Live averages (spectrum or transfer function) consist of just the name of the measurement, a round show/hide button, and a triangular run/stop button (▶). The color of the show/hide button and the border color of the control block match the display color for the associated data trace on RTA and transfer function graphs and their associated icons in graph legends. <u>*These elements are common to all types of live measurements.*</u>

To the above, we add an input level meter for single-channel spectrum measurements. Dual-channel transfer function measurements have two input level meters (labeled "*M*" and "*R*" for measurement and reference channel), a delay time field, and a delay-tracking indicator (●). You can click on the delay time field to make it editable and type in a new delay time (in milliseconds), then press the [Enter] key.

When a dual-channel transfer function measurement is the *active* measurement, an additional row of hover buttons appears below the base control set, labeled *Find, Track,* and -|+ (see Figure 63). The *Find* button starts the *Delay Finder* for the selected measurement, an automated routine for finding delay times for signal alignment (see *Activate Delay Finder* for more on this). The *Track* button turns on delay tracking, which actively re-measures and adjusts measurement signal delay on each new update of the measurement – you can also turn delay tracking on and off by clicking the tracking indicator (●). The minus and plus (-|+) buttons decrease/increase the delay time by one increment as specified in Delay options. The default is one sample (about 21 microseconds per click at 48k sample rate).

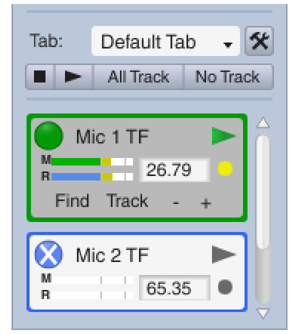


Figure 63: Control blocks for transfer function measurements.

Clicking the run button (▶) for a live measurement starts the measurement. Clicking the button again when the measurement is running stops it. The run button turns green when the measurement is running and gray when it is stopped. Note that when you start a live average, at least one of the measurements that make up the averaged measurement must also be running in order to see any data on the graphs.

Notice that when you stop a live single-channel or dual-channel measurement, Smaart automatically hides it and an "X" appears on its show/hide button. If you unhide a stopped measurement by clicking its show/hide button, you will see the last data that it acquired on applicable graphs. This can be a handy way of "freezing" a trace for closer inspection without capturing a stored data trace.

Signal Generator Controls

The next group of controls on the Control Bar is for the signal generator. The label at the top of this section is another hover button (it turns into a button when your mouse cursor passes over it). Clicking it opens the *Signal Generator* dialog, which contains a lot more options for the signal generator than we could fit on the Control Bar (see *The Signal Generator* on page 35 for more information). Below the heading are a signal type selector (*Pink Noise* is selected in the example shown here) and an output level field that shows the current output level in normalized dB full scale. The *On* button turns the generator on or off – it glows an angry red when the generator is running. The minus and plus (-|+) buttons to the right of the output level field bump the output level down or up by 1 dB.

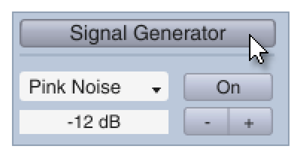


Figure 64: Signal Generator controls on the Control Bar

The default controls for the signal generator can be replaced with a more compact version by selecting *Compact Signal Generator* from the *View* menu. In the compact layout, current signal type is indicated on the button that turns the generator on and off and clicking on the numeric level readout in the center opens the *Signal Generator* control panel.

Main Display Controls

Figure 65: Main display buttons for Real-Time mode

The last group of controls in the control strip on the right side of the real-time mode window is devoted to data display functions. Starting from the top left of the screen clip shown here on the right:

- The *Spectrum* button is actually just a hard coded view preset that sets the graph area to a single pane and loads the RTA graph into it. You can also recall this view by pressing the [S] key or by selecting *Spectrum* from the *View* menu.
- The *Transfer* button is another view preset that splits the graph area into two panes and loads the transfer function Phase and Magnitude displays. The *Transfer* view preset is also accessible from the *View* menu or by pressing [T] on your keyboard. You may note that there are also 10 user-definable view presets in the *View* menu (seven of which come preconfigured by default). The *Spectrum* and *Transfer* views were given special treatment because they mimic the spectrum and transfer function modes in older versions of Smaart and SmaartLive.
- The *Live IR* button brings up the Live IR graph pane when either of the two frequency-domain transfer function graphs (Magnitude or Phase) is visible.
- The two buttons labeled with rectangles divide the main plot area into one or two graph panes: one rectangle, one pane; two rectangles, two panes. Figure 65 shows the double-pane option selected.
- The *Impulse* button exits real-time mode and switches Smaart to Impulse response mode (not to be confused with the *Live IR*). Note that in impulse response mode, the *Impulse* button changes to a *Real Time* button that brings you back to real-time mode.

❻ Command Bar

The Command Bar is a user-configurable button bar that runs across the bottom of a Smaart window. You can hide and restore it by means of the triangular button centered in the border area just above it. The show/hide button remains visible in the window border when the Command Bar is hidden and clicking this button again will restore it. You can also hide or restore the Command Bar by selecting *Command Bar* in the *View* menu or by pressing the [U] key on your keyboard. To customize the command bar, select *Command Bar Config* from the *Config* menu (see *Configuring the Command Bar* on page 38 for details).

❼ Data Bar

The *Data Bar,* which normally appears on the left side of the main Smaart window in real-time mode, is dedicated to storing and managing captured "snapshots" of live data traces (long-time Smaart users know these as "reference traces"). The data bar is essentially a window into the file folders where your captured Smaart data traces for the current active graph type are stored. The triangular button in the border between the data bar and the graph area hides the data bar – as does the *Data Bar* command in the *View* menu, or pressing the [B] key on your keyboard. The show/hide button remains visible in the window border when the Data Bar is hidden and clicking this button again when the data bar is hidden will restore it, as will the *Data Bar* menu command or keyboard shortcut.

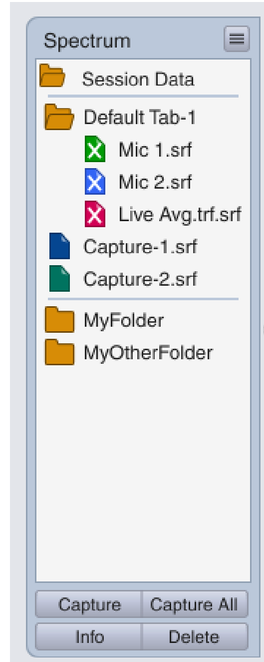


Figure 66: Data Bar for spectrum measurement data

The data bar shows you only one type of data files at a time; either spectrum or transfer function data, depending on the active graph selection in the graph area. The heading at the top of the Data Bar tells you which type of data files are currently shown. The center portion of the Data Bar lists the files and folders in your current data library folder that match the active graph type. Each file icon is colored to match the display color for the data trace stored in the file. An "X" appears in the icons for files that are currently not being displayed on the active graph.

You can organize your data files using folders and drag files from one folder to another, much as you would in any file system window on your computer, and you can drag files from the data bar onto a compatible graph in the graph area to display them. Notice that one folder is always "pinned" to the top of the library pane. We refer to this folder as the "session folder."

The session folder is the destination for new data captures and any new folders you create during your current Smaart session. You can change the session folder by clicking the (three-line) menu button in the upper right corner of the Data Bar and selecting *New Session Folder* from the menu or by dragging an existing folder to the top position in the data library pane with your mouse and then releasing the mouse button. Creating a new session folder automatically changes the session folders for both spectrum and transfer function data and the previous session folders become ordinary file folders. Dragging and dropping an existing folder to make it the session folder works similarly, except that the change applies only to the current data type.

At the bottom of the Data Bar are four buttons labeled *Capture*, *Capture All*, *Info*, and *Delete*.

The *Capture* button captures a new trace data file from the active live measurement (assuming that at least one live measurement is running). You will be asked to supply a name for the captured trace.

The *Capture All* button captures all running measurements on the active graph in the graph area. You are prompted to specify a folder name to be created in the session folder to contain the captured trace files. Each captured trace is named for the live measurement from which it was captured.

The *Delete* button permanently deletes a selected file, folder, or group of objects selected in the library pane. You can select multiple files in the library pane, either by holding down the [Ctrl/Cmd] key while clicking on file icons with your mouse (to select an arbitrary group of files), or you can hold down the [Shift] key while selecting a contiguous range of objects. Note that deleting a folder automatically deletes all of the files and folders that it contains.

The *Info* button opens the *Trace Info* dialog for a selected data file, where you can review and edit file properties. Please refer to the *Trace Info Dialog* on page 82 for more details.

Data Bar Menus

The (three-line) menu button in the upper right corner of the Data Bar opens a menu with the following commands related to captured trace data files. Note that several of these same commands are also accessible from the pop-up context menu when you right-click ([Ctrl]+click on Mac) a file in the library pane of the Data Bar. The availability of individual commands in both menus depends somewhat on your current selection(s) in the data library pane of the Data Bar.

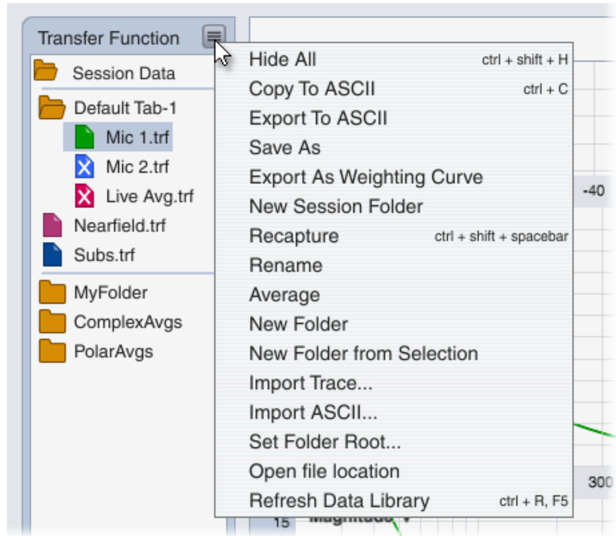


Figure 67: Data Bar Menu

The *Hide All* sets the state of all traces on the associated graph to hidden, as though you had gone through the list and clicked the icon for each visible trace to “X” them all out. This command does the same thing as the *Captured Data Traces* > *Hide All* command ([Ctrl/Cmd] + [H]) in the *Command* menu.

Copy to ASCII copies the trace data from a selected data file to the operating system’s clipboard in tab-delimited ASCII text format, suitable for pasting into a spreadsheet, text editor or any other program that accepts ASCII text. ASCII exports of spectrum data consist of two columns – frequency and magnitude – plus column headings. Transfer function traces include, frequency, magnitude, phase and coherence for each frequency. If you want to save data to a text file, you can simply paste it into a text editor such as Notepad (Windows) or TextEdit (Mac) and then save the file.

Export to ASCII exports the trace data from selected data files to tab-delimited ASCII text files. You will be prompted to choose or create a destination directory for the exported files. Exported text files are named automatically with the same names as the Smaart data traces. As with the *Copy to ASCII* function, exported data traces are created with the same smoothing or banding settings currently selected in Smaart, so what you see is what you get.

Save to File saves a copy of the selected trace to some location other than the Data Library (where it is already saved). Selecting this command opens a *Browse For Folder* dialog where you can choose or create the folder where you want to put the new copy.

Export as Weighting Curve creates a new weighting curve from a captured transfer function trace. You will then be able to apply the weighting curve to live measurements in *Measurement Config* or to captured traces in the *Trace Info* dialog – see *Weighting Curves* on page 84 for more information. This command is only present in the menu when the active graph is a transfer function *Magnitude*, *Phase* or *Live IR* graph.

New Session Folder creates new session folders for both spectrum and transfer function data files. The session folder is always pinned to the top of the data library pane of the Data Bar and it is the destination for all new captured traces and any new folders created during your Smaart session.

Selecting this command pops up a dialog box asking you for a folder name. When you click the *OK* button in the dialog, new session folders with the name that you specified are created for both spectrum and transfer function data. Your previous session folders for both data types are "demoted" to ordinary file folders in your data library.

Recapture replaces the data in the selected trace data file with fresh measurement data captured from the active live measurement on the active graph.

The *Rename* command makes the name of a selected file or folder in the data library editable, so that you can change it. Remember to press the [Enter] key on your keyboard after typing a new name to set the change.

The *Average* command does one of two things, depending on the current selection(s) in the data library pane of the Data Bar. If a folder and/or multiple trace data files are selected, Smaart offers to average the selected traces, including all traces contained in any selected folder. To multi-select files and folders in the data library, you can hold down the [Ctrl/Cmd] key on your keyboard while clicking the objects that you want to select with your mouse, or hold down the [Shift] key while selecting the beginning and end of a contiguous group of objects. If a single trace data file is selected or there is currently no selection, this command opens the full version of the *Trace Average* dialog, wherein you can select individual trace data files by clicking the check boxes next to their names. For more information on averaging traces, see *Averaging Captured Data Traces* on page 83.

The *New Folder* command creates a new empty subdirectory in your current session folder. The default name of the new folder is automatically selected for editing, so that you can simply start typing to rename it, and then press the [Enter] key to set the change.

When one or more files or folders are selected in the data library pane of the Data Bar, the *New Folder from Selection* command creates a new folder in the session folder and moves the selected objects into it in one operation. The name of the new folder is automatically selected for editing upon creation of the folder so that you can simply start typing to rename it. When you are finished editing the folder name press the [Enter] key to set the change. Note that all selected files and folders for this operation must reside either inside or outside the current session folder.

The *Import Trace* does the same thing as the *Import* > *Trace Data File* command in the *File* menu. It can be used to copy data files from other locations into your Smaart data library and can convert legacy .ref files from older versions of Smaart, Smaart Pro or SmaartLive into Smaart 8 .srf or .trf files. Selecting this command opens the *Load Reference File* dialog, where you can navigate to, and select the file(s) that you want to import. Note that this dialog only shows you native data files that match the active graph selection in the active Smaart window. If the active graph is a transfer function (magnitude, phase or live IR) graph, you will see only .trf files. If it is an RTA or Spectrograph, it will show you only .srf files. Legacy .ref files from older versions of Smaart can be either type. You cannot tell the data type by looking at the file name, so if you select a .ref file for import and nothing happens, try switching the graph type. Note that Smaart cannot open reference *group* (.rgp) files from older versions of Smaart Pro or SmaartLive. Only individual reference trace (.ref) files can be imported.

The *Import ASCII* command reads data from an ASCII text file and converts it to a Smaart spectrum or transfer function trace data file. This command works identically to the *Import* > *Import ASCII* command in the *File* menu. Please refer to *Import ASCII* on page 83 for more details.

The *Set Root Folder* command is used to specify the root directory for your Smaart data library. Selecting this command opens a *Browse For Folder* dialog wherein you can choose or create the folder where you want your Smaart trace data files to reside. Once you have made your selection, Smaart will automatically create a folder called "Traces" in the specified location (if one does not already exist) with two additional folders inside named "Spectrum" and "Transfer Function." The contents of these two folders will then appear in the data library pane on the Data Bar when a graph of the corresponding type is the selected as the active graph in the graph area.

Open File Location opens the folder where a selected file or folder resides in a standard file system window. If you move or rename any files of folders in your Smaart data library be sure to use the *Refresh* command (see below) after doing so, to ensure that Smaart picks up the change(s).

The *Refresh* command forces Smaart to reread the contents of its data library file folders. Normally, Smaart scans the library folders on startup and keeps track of changes that you make through the Smaart user interface. If you make any changes to Smaart's library folders from outside Smaart, such as adding, deleting, renaming or moving files or folders, you can use the *Refresh* command to make sure Smaart picks up the changes.

Data Bar Context Menu

The pop-up context menu that appears when you right-click ([Ctrl]+click on Mac) a file or folder in the library pane of the Data Bar is mainly a subset of the commands detailed above. There are however, two additional commands in the context menu, *Info* and *Delete Selected Trace*, that echo the functions of two of the buttons at the bottom of the Data Bar. *Delete Selected Trace* <u>permanently</u> deletes the selected file(s) and/or folder(s) – note that deleting a folder automatically deletes all of the files and folders that it contains. The *Info* button opens the *Trace Info* dialog for a selected data file, where you can review and edit file properties. Please refer to *Trace Info Dialog* on page 82 for more details.

Working with Captured Data Traces

Smaart has the ability to capture and display static "snapshots" of live spectrum and transfer function measurements as data files, for later reference. Smaart 8 can open and display captured spectrum (.srf) and transfer function (.trf) data files from version 7 or higher and can convert older .ref data files written by previous versions of Smaart going all the way back to version 1.0.

You can capture new data trace files from the active graph in the graph area by clicking the *Capture* or *Capture All* buttons on the Data Bar. The *Capture* command (keyboard shortcut: [Spacebar]) captures the *active* live measurement on the active graph. *Capture All* ([Shift] + [Spacebar]) captures *all* running measurements that match the active graph type.

Newly captured data traces are written to the designated *session folder* in your *data library*. Captured data files in the data library that are not currently displayed on a graph have and "X" drawn on their icons. To display a hidden file or hide one that is visible, click its icon. Clicking on the file's *name* selects it.

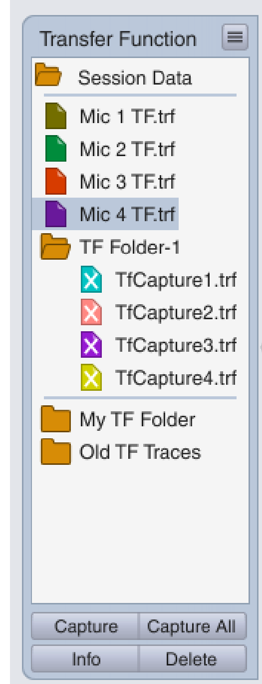


Figure 68: Data Bar for transfer function measurement data

Data Library

Smaart's data library is simply a folder on your hard disk that you designate as the location for your captured data files. The default location is a folder named "Smaart v8," located in the documents directory for your user account. In it you will find a folder named *Traces*, with two additional folders inside named *Spectrum* and *Transfer Function*. The contents of one or the other of these two sub-folders is shown on the Data Bar in Smaart when a graph of the corresponding type is selected as the active graph in the graph area.

You can change the location of your data library folder if you like, using the *Set Root Folder* command in the data bar menu. Smaart will automatically create the *Traces* folder and its two main subdirectories in the root folder location that you specify if they don't already exist. You can move files and folders in and out of these folders the same as you would any other file folders in your file system. Some things to keep in mind are that Smaart will only show you the spectrum data (.srf) files in the *Spectrum* folder and .trf files in the *Transfer Function* folder. Additionally if you make changes to the contents of any of the library folders while Smaart is running, you may need to run the *Refresh* command in the Data Bar menu in Smaart after doing so, to make sure Smaart picks up the changes.

Session Folder

One folder in the Spectrum and Transfer Function trace folders of your data library is always designated as your session folder for that data type. The session folder is "pinned" to the top of the data library pane on the Data Bar in Smaart and it is the destination for new data captures and any new folders you create during your current Smaart session. You can change the session folder by clicking the (three-line) menu button in the upper right corner of the Data Bar and selecting *New Session Folder* from the menu or by dragging an existing folder to the top position in the data library pane with your mouse and then releasing the mouse button. Creating a new session folder automatically changes the session folders for both spectrum and transfer function data, and the previous session folders just become ordinary file folders. The same thing happens when you change the session folder by dragging an existing folder to the top position, except that the change applies only to the current data type.

Organizing Your Data

You can move files around in your data library much the same way you can in a file system window, by clicking and dragging with your mouse. To create a new (ordinary) folder in your data library, click the (three-line) menu button in the upper right corner of the data bar and select *New Folder* from the menu. This will create a new folder in the current session folder, with its default suggested name highlighted and ready to edit. When you have named your new folder, press the [Enter] key to set the change. You can then drag the folder wherever you want to put it and drag files into it from other locations.

Another way to create a new folder is to select one or more files and/or folders, then either right-click in the data library pane or click the (three-line) menu button in the upper right corner of the data bar and select *New Folder from Selection*. A new folder is then created and the selected objects are moved into it. Once again, the name of the new folder is automatically highlighted for editing so that you can type a folder name of your choice, then press the [Enter] key to set the change.

To select multiple files, you can either hold down the [Ctrl/Cmd] key while clicking on file icons with your mouse to select an arbitrary group of files, or hold down the [Shift] key while selecting a contiguous range of objects. To rename a file or folder, right-click its name ([Ctrl] + click in Mac) then select *Rename* from the pop-up menu – the *Rename* command also appears in the main data bar menu but it's usually more convenient to use the right-click method. You can also rename data files in the *Trace Info* dialog.

Trace Info Dialog

The *Trace Info* dialog can be opened either of two ways: You can click the name of the file that you want to examine to select it, and then click *Info* button at the bottom of the Data Bar or right-click the file name ([Ctrl] + click in Mac) and select *Info* from the pop-up context menu.

Figure 69: Trace Info dialog for a transfer function data trace.

In the upper portion of the *Trace Info* dialog, you will find a list of everything that Smaart knows about the selected data trace and the measurement from which it was captured. This list will vary according to the measurement type but should hopefully be fairly self-explanatory.

Below the vital statistics list are editable text fields for the trace *Name* and an optional *Comment* field. Next comes the display color for the trace. You can change the display color by clicking on the *Color* tile to open a color picker dialog.

The *dB Offset* field can be used to move the selected trace up or down on applicable charts in the main chart. Positive values move the trace up on the chart by the specified number of decibels and negative numbers move it down. After editing this setting, you can click the *Apply* to see the change. Note that dB offsets are not stored in the data file. They are zeroed out each time you close Smaart.

The *Weighting* control assigns a weighting curve to the trace. Smaart has built-in curves for A and C weight curves used in SPL and Leq measurements with normal and inverted versions of each. You can also create your own weighting curves from transfer function data traces or by importing ASCII text files. See *Weighting Curves* on page 84 for more information.

The *Plot* number sets the preferred plot assignment for each trace when two charts of the same type are displayed in the main graph area. If this is set to 1 (the default), the trace will appear in the first of the two charts displayed. If it is set to 2, the trace will move itself to the second of the two charts. This setting is ignored when only one chart of a given type is displayed.

The *Save* button does the same thing as the *Save to File* command in the Data Bar menu. It opens a file dialog window, enabling you to save a copy the trace data file to any location in your file system.

The *Copy to ASCII* button copies trace data to the operating system clipboard in tab-delimited ASCII text format, suitable for pasting into a spreadsheet, text editor or any other program that accepts ASCII text.

All of the items above are common to both spectrum and transfer function traces. For transfer function traces specifically, there is one additional option and that is to display the trace upside down (*Inverted*) on the graph. This option can be handy for setting loudspeaker EQ curves.

Averaging Captured Data Traces

The *Average* command does one of two things, depending on the current selection(s) in the data library pane of the Data Bar. If a folder and/or multiple trace data files are selected, Smaart offers to average the selected traces, including all traces contained in any selected folder. To multi-select files and folders in the data library, you can hold down the [Ctrl/Cmd] key on your keyboard while clicking the objects that you want to select with your mouse, or hold down the [Shift] key while selecting the beginning and end of a contiguous group of objects. If a single trace data file is selected or there is currently no selection, this command opens the full version of the *Trace Average* dialog, wherein you can select individual trace data files by clicking the check boxes next to their names.

In either case, you are asked to name the new averaged trace and you will have the option of using either decibel (*dB*) or *Power* averaging. In the case of transfer function averages, you have the additional option of creating a *Coherence Weighted* decibel average. These are the same options available for live spectrum and transfer function averages. Please see the *Spatial Averaging* on page 19 for more details on spatial averaging options.

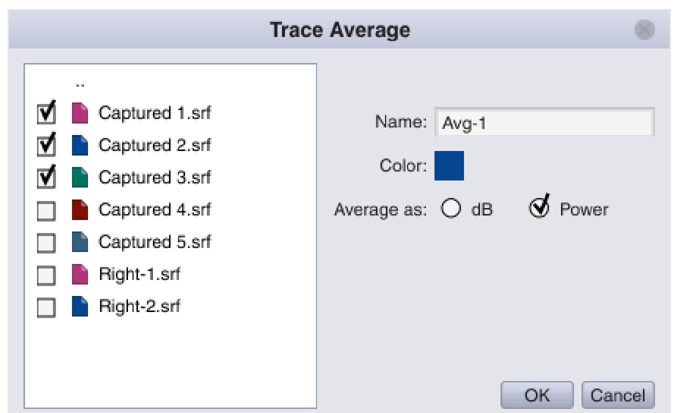


Figure 70: The full version of the Trace Average dialog.

To set the display color for the trace, click the color tile below the *Name* field to open the color picker. After making your selections, click the *OK* button. The new averaged trace data file will immediately appear in your session folder in the Data Bar and on the active graph in the graph area.

Import ASCII

In addition to its own native data files, Smaart can import frequency-domain data from ASCII text files stored in comma-delimited (.csv) or tab-delimited table format. This is a handy feature for importing spectrum or frequency response data from other programs, creating target curves or go/no-go levels, etc. To import trace data from an ASCII text file, select *Import ASCII* from the menu on the Data Bar or *Import* > *Import ASCII* from the *File* menu. Either action opens the *ASCII Import* dialog window.

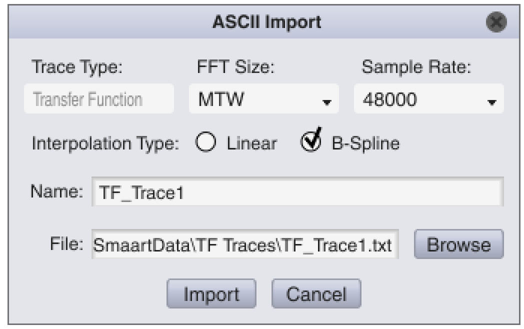


Figure 71: ASCII Import dialog

You will be creating a regular spectrum or transfer function data trace and so you need to specify an *FFT Size* and *Sample Rate* – together, these determine spacing of the frequency data points in the new trace. The current active measurement type determines the *Trace Type*.

The frequency spacing of data in the text file does *not* need to match the precise FFT bin frequencies for the selected *FFT size* and *Sample Rate*. Smaart will interpolate the frequency points that it needs from whatever set of coordinates you supply, using either *Linear* or cubic spline (*B-Spline*) interpolation. *B-Spline* is usually the better choice for real signals and smooth functions. *Linear* interpolation may work better for arbitrary curves with sharp corners. The *Name* field specifies the file name for the trace data file being created.

To select an ASCII text file for import, click the *Browse* button to bring up the *Load File* dialog, then navigate to the source file and open it. At minimum, the text file to be imported needs to have at least two sets of frequency and magnitude coordinates. Each set of coordinates occupies a line by itself with the frequency in Hertz in the first position, followed by a column separator character (a comma or tab) and a magnitude value in decibels. For transfer function traces you can add an additional column for phase in degrees. Smaart will ignore any line in the file beginning with an asterisk (*) or semicolon (;) so these may be used to add headings or comments to the file. See *Appendix G: Text File Formats for ASCII Import*, beginning on page 200 for more information.

Once all parameters are set, click the *Import* button to import data from the file. The *ASCII Import* dialog will close, a new data file is created in the session folder of your data library and your newly imported trace should immediately appear on the active graph in the graph area.

Weighting Curves

Weighting curves are used to shape the magnitude spectrum of live or stored measurement data. Another way to put it might be to say that filtering in the time domain equates to weighting in the frequency domain. Weighting curves can be applied to all real-time frequency/magnitude charts in Smaart including *RTA*, *Spectrograph* and transfer function *Magnitude* charts.

To add a weighting curve to a live spectrum or transfer function measurement, open measurement settings for the measurement –double click its control block on the Control Bar, or open *Measurement Config* and select the measurement name in the tree view. You will notice that there are *Weighting* selectors in both the individual *Measurement Settings* and the global settings sections. To apply a weighting curve globally, to all measurements of the same type that subscribe to the global setting, click the *Weighting* selector in *Global Spectrum Settings* (see page 56) or *Global TF Settings* (see page 58) and select the curve that you want to use from the list. To apply a curve locally, to just a single measure-ment, un-check the *Use Global* check box next to the *Weighting* selector in its *Measurement Settings* section (see pages 55 and 57) to enable the local control and then select your desired curve.

To add a weighting curve to a captured data trace, use the *Weighting* selector in the *Trace Info* dialog. You can open *Trace Info* for a selected file in your data library using *Info* button at the bottom of the Data Bar or right-click ([Ctrl] + click on Mac) and select *Info* from the pop-up context menu.

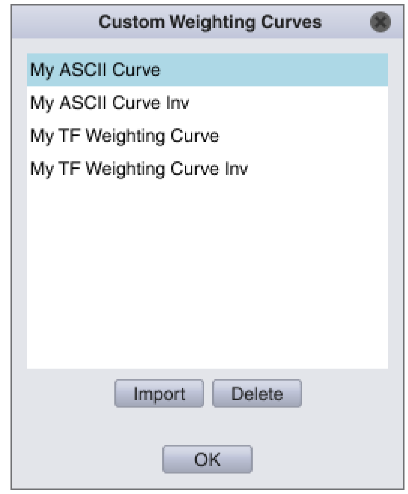


Figure 72: Custom Weighting Curves dialog

Custom Weighting Curves

Smaart has built-in curves for standard A and C weighting functions used for SPL and Leq measurements, with normal and inverted versions of each. You can also add user-defined curves, either by importing data from text files or by exporting a captured transfer function trace.

Import Weighting Curve

To import a weighting curve stored in an ASCII text file, select *Import > Weighting Curve* from the *File* menu or open the *Custom Weighing Curves* dialog (*Options* menu > *Weighting Curve*) and then click the *Import* button. Either will open the *Import Weighting Curve* dialog where you can navigate to the folder containing your custom weighting curve file and open it to import the curve data.

Weighting curve text files need to be in ASCII text format, with one frequency value in Hertz and one magnitude value in decibels per line, and a comma or tab character separating the two values. Smaart will ignore lines beginning with an asterisk (*) or semicolon (;) so these may be used to add human readable headings or comments to a file. See Appendix G, beginning on page 199 for more details.

Export Captured Transfer Function Trace as Weighting Curve

To export a captured transfer function measurement trace as a weighting curve, select the trace that you want to export on the Data Bar, then click the (three-line) menu button in the upper right corner of the Data Bar and select *Export as Weighting Curve* from the menu. A small dialog window pops up where you can specify a name for the new weighting curve. When you click *OK* in the *New Weighting Curve* dialog, the new curve is added to the *Weighting* selectors for live measurements and stored data traces and is immediately available for selection. Smaart automatically creates an inverted version of the curve as well, and appends the notation "*Inv*" to its name.

Delete Weighting Curve

To delete a custom weighting curve, select *Weighting Curves* from the *Options* menu to open the *Custom Weighting Curves* dialog, then click the name of the curve that you no longer want and click the *Delete* button. Deleting a weighting curve removes both the normal and inverted versions. Note that the standard A and C (sound level) weighting curves are hard coded in Smaart and cannot be deleted.

Quick Compare

The *Quick Compare* feature is similar to weighting, except that it is intended as a quick and temporary way to subtract one transfer function measurement from another. It is functionally equivalent to exporting a transfer function trace as a weighting curve, then applying the inverse weighting curve to all live measurements, except that there are fewer steps involved and nothing to clean up when you are finished, if you don't want to keep the weighting curve for posterity.

Capture Quick Compare

To do a quick compare you first need to copy an existing measurement into memory to use as a reference. The reference can be any live transfer function measurement or stored transfer function data trace. To capture the reference curve, make sure that the active graph is a transfer function *Magnitude* or *Phase* graph and the trace that you want to reference is the top trace on the graph. If you don't see its name showing in the upper right corner of the graph, either open the graph legend and click its name to bring it to the top of the list or just press the [Z] key on your keyboard repeatedly until its name comes up, then press [Alt/Option] + [Q] or select *Capture Quick Compare* from the *Command* menu to copy the trace into memory.

Toggle Quick Compare

When you have captured the reference curve for comparison, press the [Q] key on your keyboard or select *Toggle Quick Compare* from the *Command* menu to subtract it from all live measurements that are running. To turn quick compare off, press the [Q] key or select *Toggle Quick Compare* from the *Command* menu again. To replace the reference, repeat the *Capture Quick Compare* procedure above.

Target Curves

A target curve in Smaart is just a line with a specified spectral shape, drawn on banded RTA displays (only) at a specified level. The purpose of a target curve simply to be visible on the screen as a reference, for example when measuring background noise or tuning a system to a target response curve specified in fractional octave format, such as the cinematic X curve or noise-masking curves used for speech privacy systems.

Target curves cannot be grabbed or moved up and down on the graph as live measurement and static data traces can. They move up or down automatically to accommodate changes in banding resolution, but otherwise their position is fixed. They do not appear in the graph legend and the cursor readout ignores them. Target curves are *<u>not</u>* displayed on un-banded (narrowband) spectrum displays or transfer function graphs. However, if you need a target curve for a narrowband RTA or transfer function graph, you can make one pretty easily by importing data from a text file as a regular data trace, using the *Import ASCII* function (*see Import ASCII* on page 83).

Displaying and Managing Target Curves

To show all target curves that are currently selected for display, make sure a banded RTA graph is visible in the active tab of the active window and select *Show Target Curves* from the *Options* menu, or press [X] on your keyboard. Pressing [X] again or re-selecting *Show Target Curves* from the menu hides all target curves when they are visible.

To view and manage available target curves, select *Target Curves* from the *Options* menu (or *Command* menu) or press [Alt/Option] + [X] on your keyboard. This opens the *Target Curves* dialog. There, you can set the display status, line color and thickness for all available curves. You can also import new target curves or delete existing curves from this window.

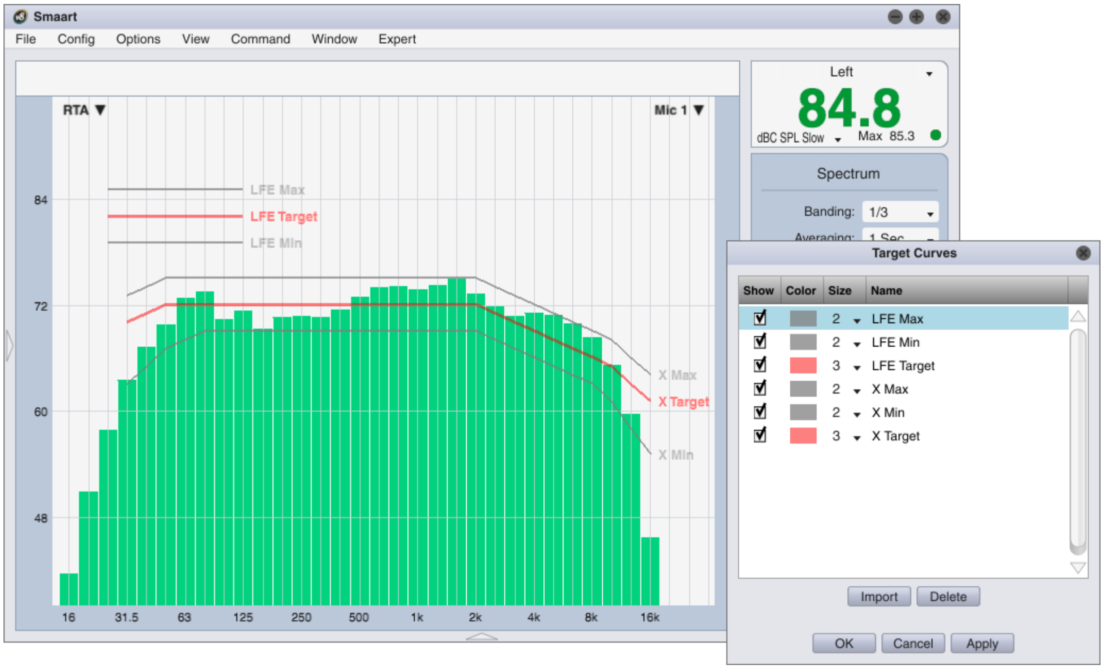


Figure 73: Target Curves on a banded RTA display and the Target Curve dialog

Clicking the check box in the *Show* column toggles the display status for each curve. When checked, the corresponding target curve will appear on banded RTA displays when target curves are turned on (see *Show Target Curves*). Clicking on any of the color tiles in the *Color* column brings up a color picker dialog, enabling you to change the display color for any curve. Clicking any entry in the *Size* column pops up a menu wherein you can specify the line thickness for each curve in pixels. Clicking the *Delete* button below the table deletes the selected curve. The *Import* button opens a file dialog where you can select a file containing a target curve specification for import. You can also import a target curve file using the *Import* > *Target Curve* command in the *File* menu.

Target Curve File Format

A target curve file is simply an ASCII text file with a special header format, followed by a list of frequency and magnitude coordinates with one frequency value in Hertz and one magnitude in decibels per line, separated by a [Tab] character or a comma.

The header format is as follows:

Line: [value]
Color: [value]
Show: [value]
Band: [value]

- The *Line* value in the header sets initial line thickness for the target curve in pixels (this setting can be changed after import in the Target Curves dialog). Allowable values are 1-5.
- The *Color* value sets line color in hexadecimal (base 16) aRGB format (alpha, Red, Green, Blue). Admittedly, this is a little clunky unless you happen to speak hexadecimal. You may want to just set the color for a new trace to FF808080 initially, for a neutral gray color, and then use the col-

or picker in the *Target Curves* dialog in Smaart to adjust line color after import. The first two characters in the aRGB string set the "alpha" value, which controls transparency. Normally these are set to "FF" (fully opaque), however you can set them to a lesser value such as 80 to get a partially transparent line.

- The *Show* value sets initial display status for the trace, where 1 means show and 0 means hide.
- The *Band* value sets the number of fractional bands per octave at which magnitude values for the target curve are specified. For example, a value of "3" means 1/3 octave banding (probably the most common choice). This setting is necessary so that Smaart can properly adjust the level of the curve on the screen when you change banding settings for the RTA display. Allowable values are 1, 3, 6, 12, 24 or 48 (full octave through 1/48-Octave).

The example shown below should produce an idealized, long-term average speech spectrum, (based on ANSI S3.5), if you were to type the text into a text file, then save the file and import it as a target curve. Remember that the frequency and magnitude values on the lines below the header would need to be separated by [Tab] characters, not spaces. Smaart picks up the file name as its curve name, so be sure to save your text file with the name that you want to appear in the target curves list.

```
Line:  3
Color: FF808080
Show:  1
Band:  3

100      55
500      62
10000    36
```

Notice that in this example, base resolution on the *Band* line is set to "3" (1/3-octave), meaning there are 20 bands between 100 Hz and 10 kHz, but we are only specifying coordinates for three points. This curve happens to be made up of two straight line segments and so we only need to specify the endpoints. Smaart will interpolate the points in between as needed.

Note also that magnitude values in this example are referenced to sound pressure level (SPL). That means that the input driving an RTA measurement in Smaart would need to be calibrated to SPL with the *Plot Calibrated Levels* option enabled in *Spectrum* options in order to display live measurement data in proper in relation to the target curve.

Network Client Window

The Client Window enables you to display and remotely control real-time spectrum and transfer function measurements running on another computer via a network connection. Supported display types include *RTA, Spectrograph*, and transfer function *Phase, Magnitude,* and *Live IR* graphs. The remote client can start and stop measurements, control averaging, banding, and smoothing (as applicable), and measure and set delays for transfer function measurements on the host. Network clients can also capture traces and control the host machine's signal generator.

Client and Server Preparation

Both the measurement server and client machine need to have Smaart 8 installed and both copies should be up to date with the latest software revisions. All tabs and measurements required by the client need to be configured in a single Smaart window prior to connecting and no changes should be made to this window while the client is connected.

The remote client window connects to a selected Smaart window on the server machine and reads all of its tabs and measurements upon connection. The client cannot switch windows once connected and will not be able to detect any changes to tabs or measurements in the window that it connects to without disconnecting and reconnecting.

Figure 74: The API page in the Options dialog

Smaart can serve data to multiple clients simultaneously but clients cannot detect changes to measurement settings made at the server or by other clients. For example, if two clients are connected to the same host and one of them changes the averaging setting for a measurement, that change will affect the measurement data that both clients see, but only the client that modified the setting will see that it was changed.

You may therefore wish to dedicate a window on the server machine to each remote client and keep those windows minimized on the server to keep them out of harm's way. This is an especially good idea when an operator is actively using Smaart on the host machine while remote clients are connected. In cases where just one person is using both machines, such measures may not be necessary.

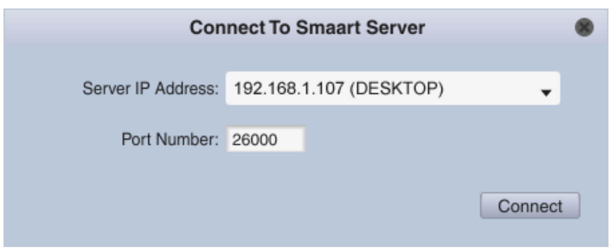
Network Configuration

We recommend that at least one of the machines in a client-server relationship be hardwired to the network. If both are operating on wireless connections, performance may suffer. Configuring the server to accept network client connections is a just matter of enabling Smaart's network API in *API* options (*Options* menu > *API*). The default *Port* address of 26000 should work fine for most purposes. You would generally need to change that only in the event of a conflict with some other application using the same port address on the same Ethernet adapter, but conflicts should be rare. The *Password* is optional. If you leave this field blank, no password is required to connect. *Show Password* makes the *Password* field readable.

The first time that you enable the API, you may get a warning from your operating system that Smaart is attempting to access the network, asking if you want to grant permission. Be sure to tell it yes if this happens, so that Smaart does not get blocked by your firewall. You may also need to reauthorize access in some cases after installing software updates.

On the client side, it's a good idea to *disable* the network API to prevent another copy of Smaart from trying to connect to that machine while it's operating as a client – any installation of Smaart can be a client *or* a server, but not both at the same time. To connect a client to the server machine, select *Client Window* from the *View* menu on the client machine or use the keyboard shortcut [Alt/Option] + [R]. You will be presented with a connection dialog like the one below. Here again, if you are doing this for the first time, the OS may warn you that Smaart is trying to access the network and ask if that is OK. Be sure to say yes, if so.

If client and server machines are both connected to the same local area network (LAN), then Smaart should be able to auto-detect the server's socket address (the IP address and Port number). If there are multiple Smaart servers available on the same LAN, you can click the down arrow to the right of the *Server IP Address* field to see a list and select the one that you want. If the server that you want to connect to is *not* on the same local network as your client machine (or perhaps on a different subnet), you can enter its IP address and port number in the fields provided. After editing each field, remember to press the [Enter] key on your keyboard to set the change.



When you have made your selections, click the *Connect* button to proceed. If Smaart connects successfully you will be asked for the password if applicable and then, if multiple Smaart windows are open on the server, you will see another dialog asking you to choose which window you want to emulate. If there is only one window open on the server, Smaart will just connect to it and go on. Once you have established your server connection and window selection, the client window will open.

Client Window Usage

The client window looks very much like any other Smaart application window (with all tabs in Real-Time mode) and operates much the same way, although there are some limitations. The most immediately visible differences are that the SPL Meter/Clock display above the control bar is replaced with a block of information about your server connection. The *Impulse,* and Measurement Config (hammer and wrench) buttons on the Control Bar are disabled, as are some menu selections. Peak hold, and coherence blanking are currently unsupported, as are unwrapped phase, phase-as-group-delay and SPL metering.

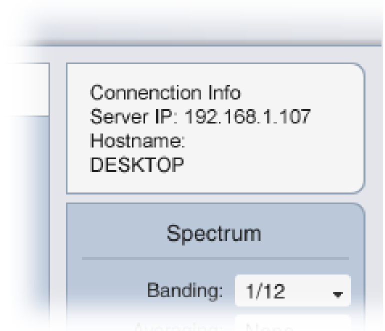


Figure 75: SPL Meter/Clock in the client window is replaced with network connection information.

On the *Data Bar*, the *Data Library* pane shows your local repository. Captured traces assigned to graphs in the window being emulated on the host machine are *not* carried over to the client when you connect. When you capture a trace in the client window, however, Smaart actually captures the file on the host machine and then uploads a copy to the client, meaning the captured data file will then exist on both machines.

If you can watch both the host and client windows at the same time, you will see that some changes to measurement settings in the client window affect the corresponding settings on the server, and some do not. *Averaging*, for example, is done at the measurement engine level in Smaart and therefore must be done on the server. *Banding* and *Smoothing* are post-process display functions and are done on the client end; the server and client can have different settings for these.

Client Window Settings in API Options

The following settings are found in the Client Window settings section of the *API* options dialog page (see Figure 74 on page 89). These are sent from the *client* machine to the API server upon connection. To access the *API* options page, select *API* from the *Options* menu .

- *Spec / TF Stream FPS* sets the frame rate for real-time spectrum and transfer function measurements in frames per second. This is set to the maximum allowable value (23 FPS) by default. You can set this to a smaller number to slow down the client's refresh rate if you need to reduce the bandwidth requirement for your network connection.
- The *Command Timeout* setting adjusts the amount of time in milliseconds before an API command is abandoned by the server. This value should not be increased from the default setting of 2000 ms unless you are experiencing problems with API commands failing.
- *Live IR Range* sets the total time range for the transfer function Live IR display in the client window in ms. For example, a setting of 20 ms will send Live IR data in the range of -10 to +10 ms, relative to time zero. Note that larger settings increase the bandwidth requirements for network connections.

Chapter 5: Single Channel Measurements

Spectrum Measurement and Display Configuration

Single-channel spectrum measurements allow you to examine the spectral content of audio signals throughout a system. Spectrum measurements are extremely useful in many applications, including the identification of feedback frequencies in sound reinforcement systems, noise and sound exposure measurements, and general signal monitoring tasks. Data from spectrum measurements can be displayed as a conventional RTA (real-time spectrum analyzer) graph, or plotted over time in a three-dimensional (level vs. frequency vs. time) Spectrograph chart.

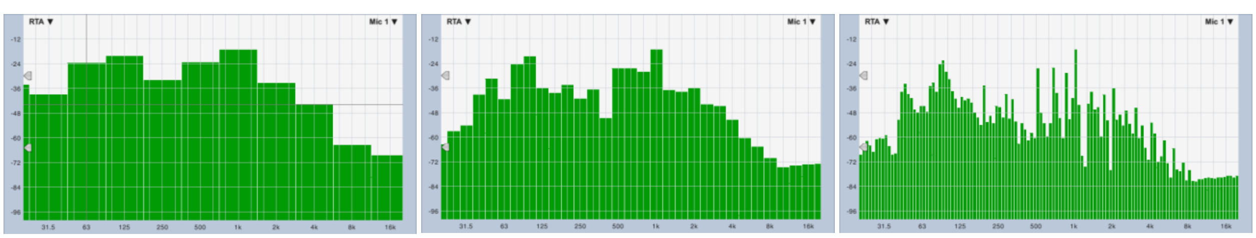
Two basic groups of settings determine the appearance and behavior of RTA and Spectrograph displays in Smaart:

- Measurement settings affect how data is acquired. These, we have discussed in detail in Chapter 3. In this section, we will look more specifically at how some of those options affect the *RTA* and *Spectrograph* displays.
- Display settings affect how spectrum measurement data is displayed after it is acquired but do not change the underlying measurement data. These options mainly reside on the *Spectrum* page of the options dialog (*Options* menu > *Spectrum*), which we will be talking about in this chapter.

In practice it isn't really possible to draw a completely hard line between the two – for example, fractional octave banding is actually done at display time but we treat is as a measurement parameter for practical reasons – but that is the basic organizational intent, in terms of where the various options for spectrum measurements are located in Smaart.

RTA Measurements

The real-time spectrum analyzer, or *RTA*, is a familiar tool to most audio professionals and probably needs little introduction. It enables you to look at the frequency content of signals moment-by-moment in real time. Essentially the *RTA* is a graph of the energy in an incoming signal, broken down by frequency or frequency ranges, with frequency (in Hertz) on the *x* axis and magnitude (energy) on the *y* axis in decibels (dB). The graph is updated continuously when one or more live spectrum measurements are running, to produce a real-time display. By adjusting the scale and averaging of the display, we can refine measurement resolution and responsiveness to fit different tasks.



Octave banding | 1/3rd Octave banding | 1/12th Octave banding

Figure 76: RTA display with Octave, 1/3-octave and 1/12-octave banding.

Time and frequency resolution are major trade-offs associated with spectrum measurements and real-time frequency-domain analysis in general. We always have to give up a little of one to get more of the other. On a basic level, the FFT size used to transform time-domain signals into frequency-domain spectral data limits the time and frequency response of RTA and Spectrograph. Larger FFT sizes give you tighter frequency spacing and more detail at low frequencies at the expense of integrating over a longer period, which can limit your ability to see fast changes in the signal. The other major factors affecting the degree of detail that you can see on the RTA graph and its responsiveness to changes in the input signal are averaging and banding.

Averaging and Banding Controls

For the RTA display, we typically average data from successive incoming FFT frames over some period of time to produce a display that is smoother and less jumpy. However averaging also limits how quickly the RTA can respond to rapid changes in the frequency content signals. Unlike banding, FFT size and averaging are baked into RTA data at the measurement level – when you capture an RTA trace you are capturing averaged data (if applicable). Note that spectrum averaging affects only RTA data. The Spectrograph is always plotted from instantaneous (un-averaged) spectrum data.

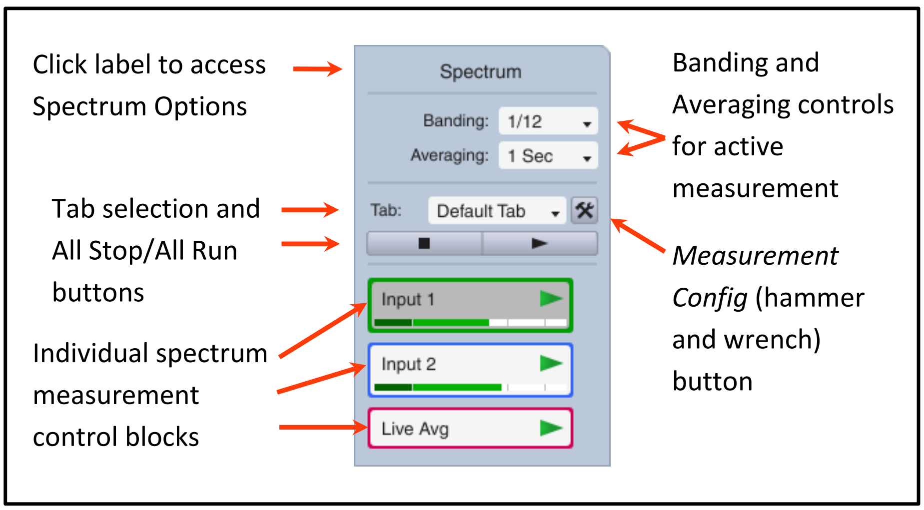


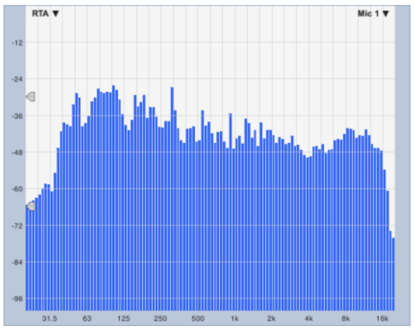
Figure 77: Live measurement controls for spectrum measurements on the Control Bar.

Banding is actually a display parameter for Spectrum data. When you capture a snapshot of an RTA trace, you are capturing it at the original FFT resolution and can always change the banding after the fact. Using larger fractional octave bands can help reduce visual "noise" and make larger trends in the spectrum of a signal more obvious, but at the expense of limiting how much detail you can see.

When the active display is an *RTA* graph, the *Banding* and *Averaging* settings for the active spectrum measurement are adjustable from the Control Bar on the right side of the main Smaart window. *Banding* is a global setting that applies to all spectrum measurements. Averaging applies only to the *RTA* display and can be set specifically for individual spectrum measurements. If the current active measurement uses the global setting for spectrum averaging then the *Averaging* selector on the Control Bar controls the global setting. Otherwise, it applies only to the active measurement.

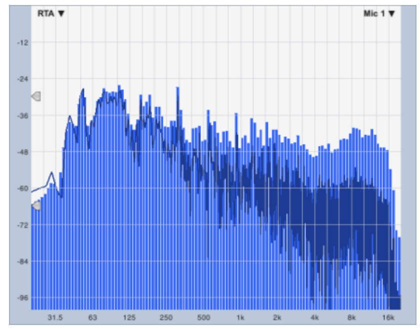
RTA Graph Types

Most commonly, the *RTA* is displayed as a bar chart with fractional-octave frequency scaling, however Smaart can display banded RTA data as a line graph, or plot a combination of fractional octave data in bar chart form with the un-banded FFT data overlaid as a line graph. Un-banded (aka "narrowband") FFT data is always plotted as a line graph. These three options for *Banded Data* (*Bars, Lines* or *Both*) are located in the *RTA Display Settings* section of the *Spectrum* page in the options dialog window (*Options* menu > *Spectrum*).



Fractional-Octave Bar Graph (*Bars*)

Fractional-Octave Line Graph (*Lines*)



Fractional-Octave Bar Graph and Unbanded Line Graph (*Both*)

Figure 78: RTA graph types (Bars, Lines or "Both")

Peak Hold

When looking at dynamic signals on an *RTA* display the normal bar or line graph is typically averaged over some period. Without averaging, the display can be too jumpy to read, but averaging tends to smooth out some of the faster peaks in the signal. If you want to see both averaged power and a record of the highest level the peaks in the signal at each frequency or in each band, you can turn peak hold on and off by selecting *Toggle Peak Hold* in the View menu or pressing the [P] key on your keyboard. Peak hold data is plotted as a second line trace on line graphs or as a series of flattened bar segments on bar charts. When you capture an RTA trace with peak hold turned on, both the normal RTA trace and the peak hold data are stored in the captured measurement.

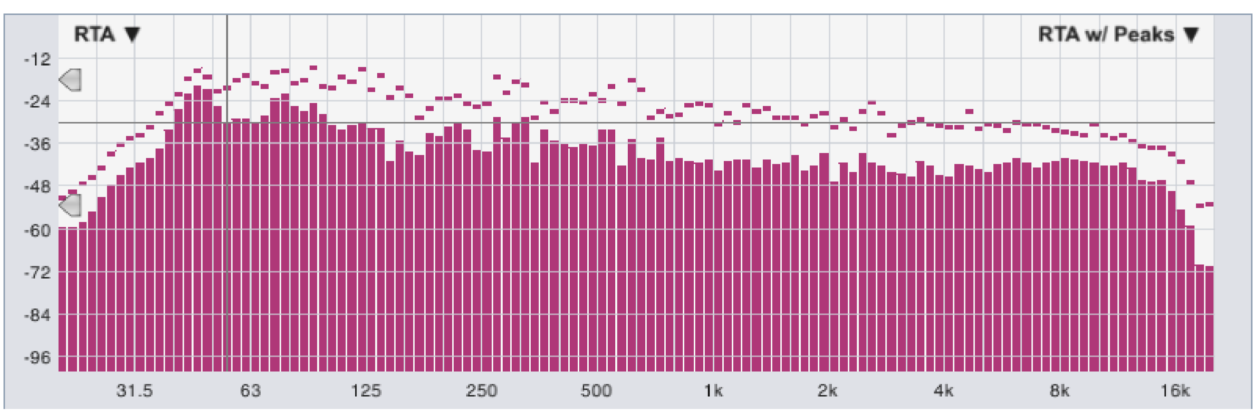


Figure 79: RTA bar graph with peak hold

The *Peak Type* selector in the *RTA Display Settings* section of *Spectrum* options (*Options* menu > *Spectrum*) sets the peak hold type. The choices are *Infinite* or *Timed*. *Infinite* peak hold preserves the highest peak level recorded for each frequency until it is either replaced by a higher reading or you turn the feature off or press the [V] key to flush the averaging buffer. *Timed* peak hold allows the peak trace to decay after some period of time, as specified in the *Hold* field.

By default, the peak hold function looks for the highest peaks in each incoming FFT, before the data goes into the average for the normal RTA display, meaning that you may never see an averaged RTA trace come anywhere near the peak levels. If you want to look at the highest levels in the averaged signal instead, click the *Averaged* check box in the *RTA Display Settings* section in *Spectrum* options. When using this option, you may want to run the *RTA* for a few moments and give the live measurement(s) a chance to settle before turning on peak hold.

Plot Calibrated Levels

The *RTA* display in Smaart is calibrated to digital Full Scale by default, meaning that the largest magnitude value obtainable in a digital sinewave (given the current *Bits per Sample* selection in *I-O Config*) is scaled to 0 dB and all lesser magnitudes end up being negative decibel values. This works very well unless you need to relate the *RTA* display to some external reference, such as sound pressure level (SPL).

If one or more of your spectrum measurements uses an input that is calibrated to SPL (which we will get to later in this section), and you want the *RTA* display to reference the calibrated levels, open *Spectrum* options (*Options* menu > *Spectrum*) and click the *Plot Calibrated Levels* check box in *RTA Display Settings* to enable it. This applies the specified input calibration offset for each input channel to spectrum measurements before plotting *the* RTA graph. It also sets the default RTA magnitude range to 20 dB to 120 dB (rather than 0 to -100 dB).

One potential problem you may encounter when using the *Plot Calibrated Levels* option is that it can result in a drastic difference in scaling between calibrated data inputs and uncalibrated measurements; for example, between calibrated microphone inputs and line level inputs. Uncalibrated spectrum measurements tend to “fall off” the graph when *Plot Calibrated Levels* is enabled. The best way to work around this issue is to add a dummy calibration offset to uncalibrated inputs channels used for Spectrum measurement, so that they rescale themselves along with calibrated measurements when the *Plot Calibrated Levels* option is selected.

To assign a dummy calibration to an uncalibrated input, first bring up an *RTA* graph and run all spectrum measurements with *Plot Calibrated Levels* disabled, to make sure all measurements are visible on the graph and have signal present. With the RTA still running, turn on *Plot Calibrated Levels* in *Spectrum* options, then select *I-O Config* from the *Config* menu or press [Alt/Option] + [A]. In *I-O Config* select the input device that is driving an uncalibrated measurement in the devices table on the upper left to display its input channels in the channels table below. There you will see that uncalibrated inputs have a *Cal. Offset* value of 0.00 dB. You can click on any entry in the *Cal. Offset* column to edit this value, and then press the [Enter] key to set the change.

If you are trying to match a line input to a microphone input that is calibrated to SPL, an offset of 120 dB is generally a good place to start. After changing the calibration offset, click the *Apply* button in the lower right corner of the dialog window and hopefully your measurement will appear on the RTA graph. If you want to move it up or down, you can adjust the offset value and reapply the change.

Spectrograph Basics

Smaart's Spectrograph straddles the time and frequency domains giving you a birds-eye view of the frequency content of a signal over time. The real-time spectrograph and the IR mode version are essentially the same display, oriented in two different directions.

If you are new to the spectrograph, then one way to think about how it works is to start with the idea of a spectrum analyzer. On a real-time spectrum analyzer (RTA) you typically have a bar graph or line chart with frequency on the *x* axis and magnitude in dB on the *y* axis, showing you the spectrum of some chunk of signal at a given moment in time – perhaps something like the one shown in Figure 80a.

An RTA is very useful tool, but if you want to get a better understanding how the spectrum of a signal changes over time, you either need a really good memory or a different kind of graph. One solution might be to just keep sliding the old data to the back instead of erasing it as new data comes in, to form a 3-D graph with time on the *z* axis, as in Figure 80b. If you did this with a 3-D area chart instead of a bar graph you would have what is commonly called a waterfall chart, but let's continue with the bar graph example, as both have the same limitation. The problem with this approach is that as new data comes in, higher-level values in front will cover up some of the data in back, so that you only get a partial picture of the history, as in Figure 80b.

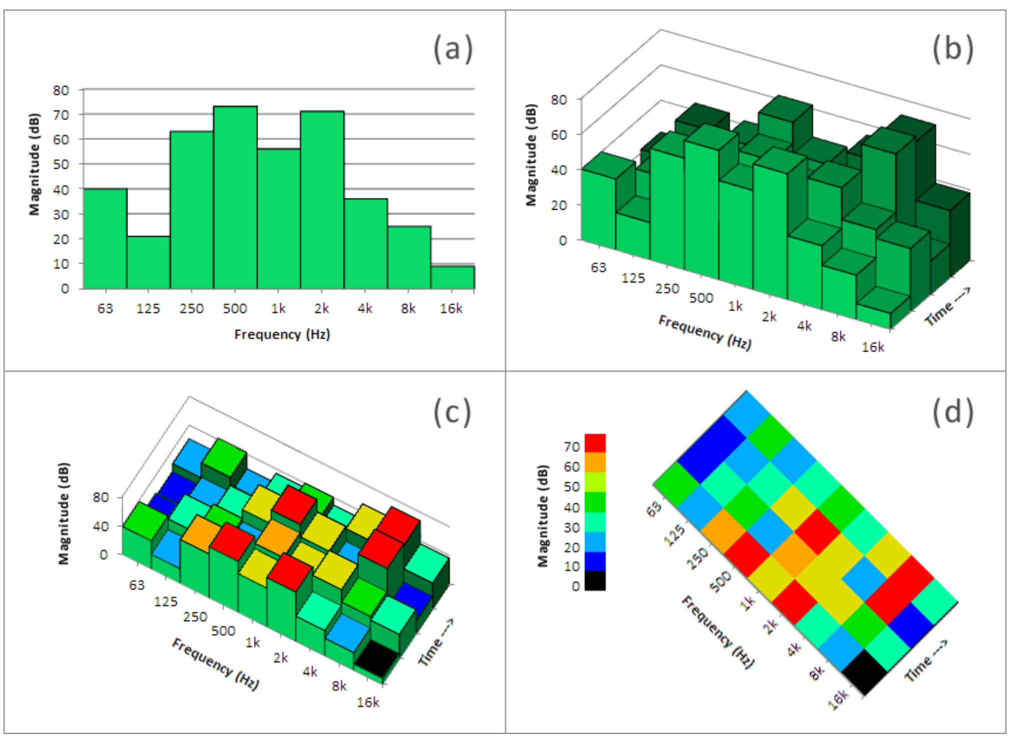


Figure 80: Turning a spectrum analyzer into a spectrograph

You *could* rotate the graph in space until you can see all of the bar tops, but when you do that it becomes harder to discern how tall they all are. Assuming that you have a color display (waterfall charts were popular before anyone had color monitors) you might try to alleviate that problem by painting the tops of the bars different colors based on their relative magnitude as we did in Figure 80c. But at that point, the chart would much more readable if you just dispensed with the bar graph idea altogether and plotted it as a 2-D chart instead, with frequency on one axis, time on the other, and magnitude indicated by color (Figure 80d). That's a spectrograph.

Generally, the "domain" of a graph is the independent variable, e.g., time or frequency, which is normally assigned to the horizontal (x) axis, but the spectrograph display has two independent variables. You can orient it whichever way is most convenient. In real-time mode in Smaart we want to relate the spectrograph to other frequency-domain graphs, so we plot it with frequency on the *x* axis and time on the *y* axis. In IR mode, we most often want to look at it in the context of other time-domain graphs, so we do it the other way around – in that case, time goes on the *x* axis and frequency on the *y* axis.

The Real-Time Spectrograph

Now that we have a general idea of how a spectrograph works and what it can tell us, let's look more specifically at the real-time spectrograph. Smaart's real-time spectrograph display is a plot of a signal's spectrum over time, with frequency (in Hertz) in the *x* axis, time on the *y* axis, and magnitude in decibels represented in color. The time axis of the graph is unreferenced, because the update frequency can vary somewhat, depending on how busy your computer is at any given time.

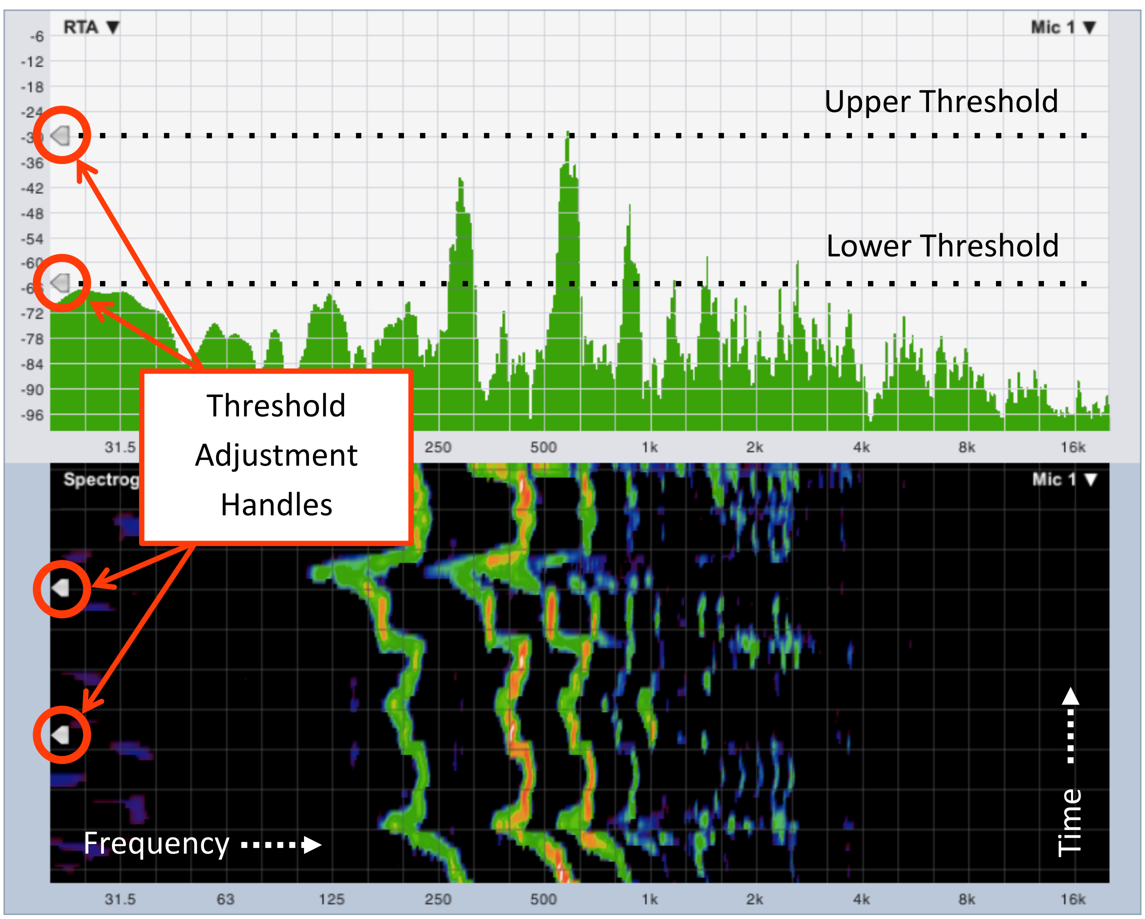


Figure 81: The real-time mode Spectrograph display

Spectrograph Dynamic Range

The dynamic range of the spectrograph is controlled by two arrow-head-shaped widgets that appear on the left edge of the Spectrograph chart. These controls also appear on the RTA graph. The upper of the two widgets sets the maximum (*Max*) threshold; the lower one sets the minimum (*Min*). The spectrograph scales its color spectrum between these two extremes. Any FFT bin whose magnitude exceeds the specified maximum is mapped to the color white. Values falling below the minimum are mapped to black. Note that you can also specify threshold values for spectrograph dynamic range in the *Spectrograph Settings* section of *Spectrum* options (*Options* menu > *Spectrum*).

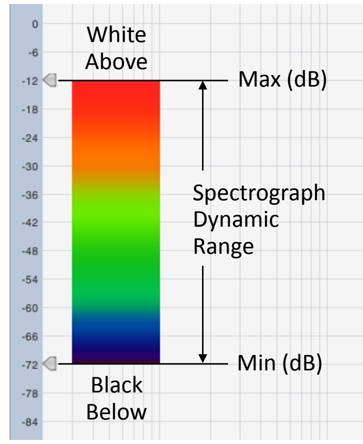


Figure 82: Spectrograph dynamic range and color mapping

The real key to creating a useful spectrograph is getting the dynamic range right. If you set the range too wide, the display loses definition and important features may get lost. Set it too narrow or set the lower threshold too high and you might miss some important features altogether. One of the major advantages of Smaart 8's real-time spectrograph is the ability to adjust spectrograph thresholds dynamically, so that you can see the affect that adjusting the Min/Max thresholds has on data already on the screen and make an "apples-to-apples" comparison.

Buffer Size and Slice Height

Smaart can maintain a fairly lengthy history for the Spectrograph. The amount of spectrograph history that you can *display* still limited by screen resolution, graph size, and the spectrograph slice height, but you can keep as many as 2000 slices of history in memory. That works out to at least 80 seconds worth of data at a maximum update speed of 24 frames per second. When the history size exceeds the graph size, you can scroll the graph backward and forward using the up and down arrow keys on your keyboard (assuming that a spectrograph is the active graph).

The *Slice Height* setting in the *Spectrograph Settings* section of the *Spectrum* options dialog determines the vertical height of each horizontal "slice" of spectrograph. The smaller the slice height, the more data you can fit on the screen. Larger slices may make small features easier to see and may also consume fewer graphics processing resources. Note that you can also change the slice height for the Spectrograph from the main window, using the plus and minus ([+] / [−]) keys on your keyboard, when the active graph is a Spectrograph.

Grayscale

One additional option for the real-time spectrograph is to render the graph in grayscale, rather than in color. Users who are colorblind or otherwise have trouble distinguishing between some colors may find the grayscale spectrograph easier to read than the color version. The grayscale option may also produce better results when making screen shots for printed documents, if you don't plan on printing the document in color. To change the spectrograph to grayscale, open *Spectrum* options (*Options* menu > *Spectrum*) and click the check-box labeled *Grayscale* in the *Spectrograph Settings* section.

Spectrum Options

We have already covered the use of most settings in *Spectrum* options but not quite all. To pick up the ones that we have not talked about and recap the ones that we have, here is a full list of all the settings on the *Spectrum* page of the *Options* dialog.

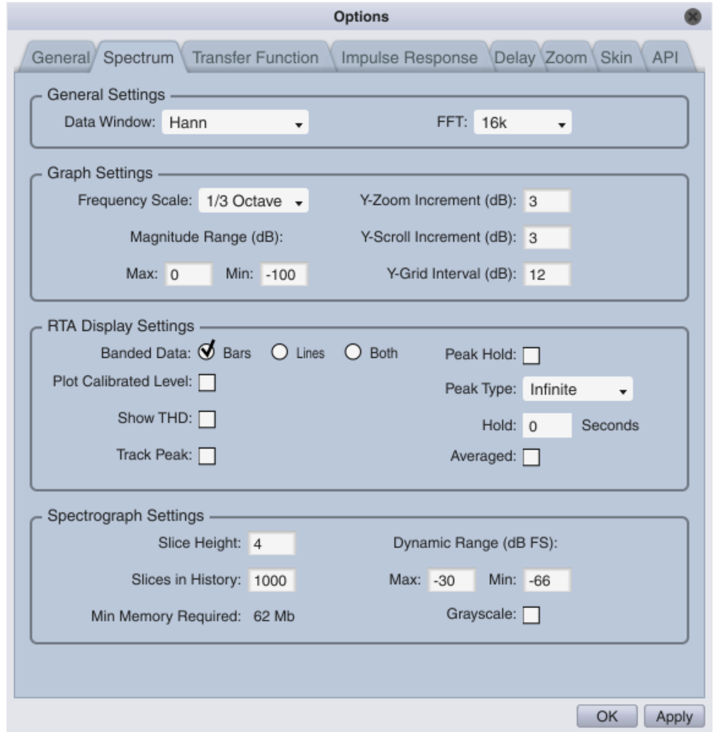


Figure 83: The Spectrum options dialog tab

General Settings

Data Window sets the data window function, used to precondition time-domain signals before performing an FFT for narrowband spectrum measurements (banded spectrum measurements always use a Hann window). The default setting is Hann, which is generally a good place to leave it unless you have some good reason to change it.

The *FFT* selector sets the global FFT size (in samples) for all spectrum measurements. The FFT size, along with sampling rate, determines the time and frequency resolution of the measurement. Given a sampling rate of 44.1k or 48k samples per second, the default setting of 16K points is generally a good trade-off that works well for most audio applications. The functional equivalent for 88.2k or 96k sampling rates would be 32K.

Graph Settings

The *Frequency Scale* selector sets the frequency scale and grid ruling options for the RTA display. There really are only two actual *scaling* options: linear (*Lin*) and logarithmic. All of the options in this list other than *Lin* are grid-ruling options for logarithmically scaled frequency.

- *Decade* plots the RTA graph with logarithmic frequency scaling and decade (base 10) vertical grid ruling.
- *Octave* plots the RTA graph with logarithmic frequency scaling and vertical grid lines spaced at one-octave intervals.
- *1/3 Octave* plots the RTA graph with logarithmic frequency scaling and vertical grid lines spaced at 1/3rd-octave intervals.
- *Lin* plots the RTA graph with linear frequency scaling and linearly-spaced vertical grid ruling.

Magnitude Range (dB) sets the decibel range for *y* axis of the RTA graph. Note that the default *y* ranges (calibrated and un-calibrated) for RTA graphs are hard coded. These settings control only the current display range and are overwritten if you zoom the plot using mouse or keyboard shortcuts or reset to the default range.

Y-Zoom Increment (dB) sets the increment used for keyboard zoom in on the *y*-axis of the RTA graph. When an RTA display is selected in the plot area pressing the [+/=] or [-] keys will increase or decrease the vertical scale of the graph by the number of decibels specified here.

Y-Scroll Increment (dB) sets the increment for keyboard scrolling on the RTA or Spectrograph displays. When an RTA or Spectrograph display is selected in the plot area, each press of the up arrow [↑] keys will scroll the plot up or down by the number of decibels specified.

Y-Grid Interval (dB) sets the *y*-axis grid-ruling interval for the RTA graph in decibels.

RTA Display Settings

The *Banded Data* setting in the *RTA Display Settings* section of *Spectrum* options determines the type of chart used for octave and fractional octave band displays.

- Selecting *Bars* plots banded RTA graphs as a bar chart.
- Selecting *Lines* changes the banded RTA display to a line graph.
- The *"Both"* option is a hybrid display that superimposes a narrowband spectrum trace over a fractional-octave banded bar graph.

Plot Calibrated Level applies input calibration offset (e.g., for SPL calibration) to RTA measurements (if applicable) and makes the default RTA display range 20 dB to 120 dB. Note that this will result in a drastic difference in scale between data from inputs calibrated to SPL and inputs calibrated to digital Full Scale. When this option is not selected, Smaart ignores input sensitivity calibration and references all spectrum measurements to digital Full Scale. The default magnitude range for RTA displays in that case is 0 to -100 dB.

When *Show THD* is enabled and the RTA graph is set to a fractional-octave resolution of 1/12th-octave or higher, the notation THD: n%, will appear in the cursor tracking readout above the RTA graph, where n is the THD percentage value calculated for the current cursor frequency. THD values in Smaart are the ratio of the power in the fractional octave band at the cursor frequency, to the sum of the power in the next three harmonic frequencies. ***If (and only if)*** the cursor is positioned at the frequency of a single sine wave being used to stimulate the system under test, this value should be indicative of the total harmonic distortion present in the system at that frequency. Otherwise, it is generally meaningless.

Track Peak causes Smaart to track and display magnitude and frequency of the data point with the highest magnitude in the front trace on the RTA plot when enabled.

The *Peak Hold* check box turns peak hold on.

Peak Type sets the time decay type:

- *Timed* peak hold allows the peak trace to decay after some period of time, as specified in the *Hold* field.
- *Infinite* peak hold preserves the highest peak level recorded for each frequency until it is replaced by a higher reading or you turn the feature off of press the [V] key to flush the averaging buffer.

The *Hold* field sets decay time for *Timed* peak hold in seconds.

Averaged peak hold bases the peak hold trace on averaged RTA data when selected (if applicable). Otherwise, the peak hold function looks for the highest peaks in each incoming FFT, before the data goes into the average for the normal RTA display.

Spectrograph Settings

Slice Height sets the height in pixels for each row in the Spectrograph display.

Slices in History sets the maximum number of rows for the spectrograph history. This can exceed the number of rows you are able to display on your screen at a given time, allowing you to scroll back to see transient events or other features that have scrolled off the screen. The caveat is that the more rows in the history, the more memory is required. For large FFT sizes and very long histories the memory requirements can be quite large

Max Memory Required shows you memory requirements for the specified spectrograph history size based on the current FFT size selected for spectrum measurements in *I-O Config*.

Grayscale changes the spectrograph to shades of gray, rather than colors.

Dynamic Range (dB FS) sets the upper and lower (*Max* and *Min*) boundaries for the spectrograph display in decibels. Frequency data points whose magnitude values fall below the specified minimum (*Min*) value are displayed in black. On the high end, out-of-range values that exceed the specified *Max* value are set to white.

Application Examples

What follows are some examples of spectrum measurements used in "real world" applications. These are intended as walk-though exercises that you can use to get a little hands-on practice, as well as examples of useful things that you can do with Smaart.

Distortion and Overload

For this exercise, we will start with a very simple measurement setup and provide detailed instructions for every step in the process as though you were starting from scratch with a new configuration. If you already know your way around Smaart then just skip over parts that you already know.

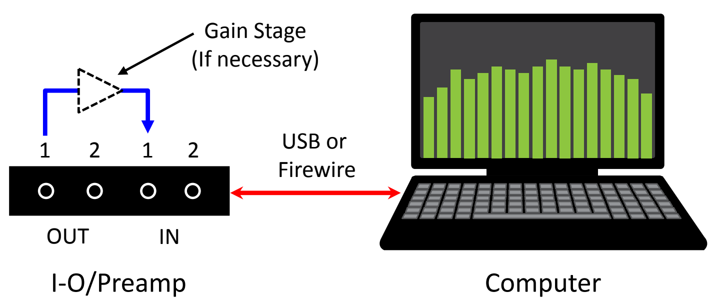


Figure 84: Loopback setup for distortion and overload example

If the headphone output on your computer is capable of overloading its line inputs, the measurement setup for this example may be as simple as a 3.5 mm (1/8") TRS patch cable. Otherwise, you may need to route the output of your audio I-O device through a gain stage of some kind, such as a mixer or preamp, and then back into an input. The idea is that we want to be able to drive a sinewave hard enough to clip the input on your audio I-O device.

Click the *Spectrum* button at the bottom of the Control Bar to ensure the graph area has a single graph pane with the graph type set to RTA. If you don't already have a spectrum measurement set up for the input device and channel that you will be using, select *I-O Config* from the *Config* menu or press [Alt/Option] + [A] on your keyboard to open the *Configurator* dialog to the *I-O Config* page.

In *I-O Config*, make sure *Input Devices* are selected at the top of the page then select the *Use* check-box next to the input device that you will be using. Smaart will automatically select all of its channels for use in the channels table below and create a spectrum measurement for each channel with the *Friendly Names* from the channels table as their measurement names. If you want to change the name of an input and its associated spectrum measurement, click the on the *Friendly Name* for the channel and type a new name, then press the [Enter] key to set the change. If there are any channels on the device that you do not want to use, you can un-check their *Use* check boxes in the channels table.

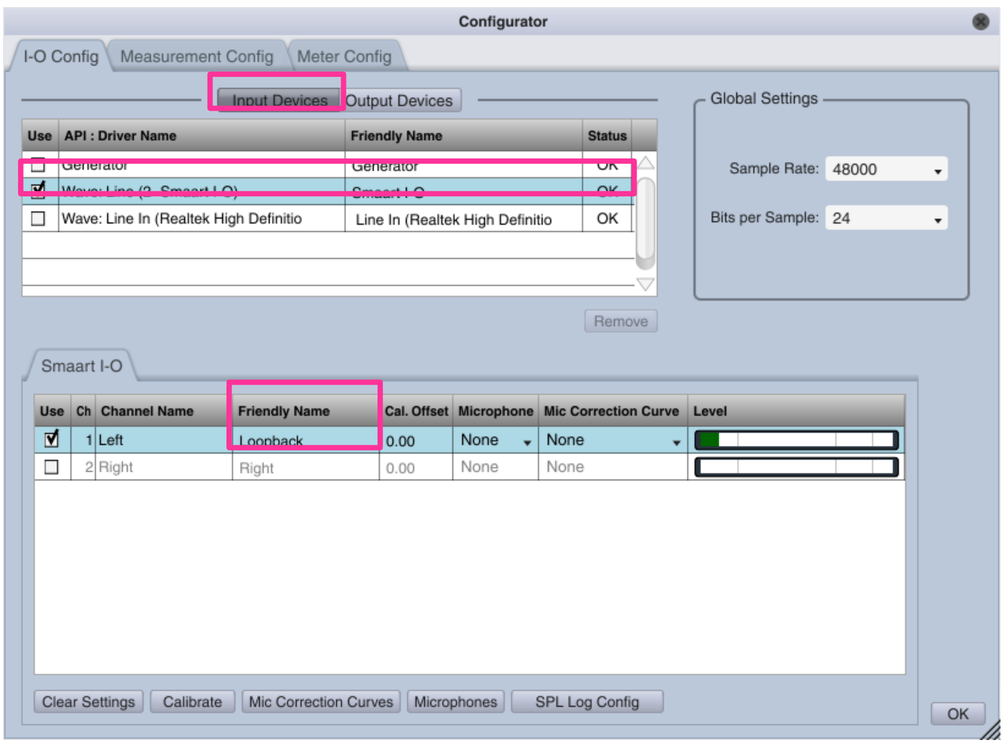


Figure 85: I-O Config setup for distortion and overload example application

When you have made your selections, click *OK* to exit the Configurator dialog. Back in the main Smaart window, you should now see a control block for your new measurement in the lower portion Control Bar. Click the run (▶) button for the measurement to make sure it works. All we care about at this point is making sure the run button turns green and Smaart does not throw any error messages.

If there's any problem, go back into *I-O Config* and make sure the status of the input device that you selected is showing "OK." If it says "N/C", then the device is either not connected or there is some kind of hardware or driver problem, so choose another device or stop and troubleshoot the problem.

Once you have confirmed that the measurement is working, set the *Banding* selector on the Control Bar to 1/48 Octave and the *Averaging* to 8 FIFO. If you have changed the FFT size for spectrum measurements, click the Spectrum label at the top of the Control Bar and change it back to the 16K in *Spectrum* options. Click on the heading of the *Signal Generator* control group on the Control Bar or select *Signal*

Generator from the *Options* menu to open the *Signal Generator* options dialog. In the dialog window, select *Sine* as your *Signal* type, then set the master *Level* value to -12 dB and the *Level 1* gain to 0 dB. Set the frequency (*Freq*) to 1000 Hz and then click *OK* to exit the dialog.

The signal type selector in the Signal Generator control group in the Control Bar should now say *Sine* (instead of *Pink Noise*) and the output level should be -12 dB. Go ahead and click the *On* button with your mouse or press [G] on your keyboard to start the signal generator.

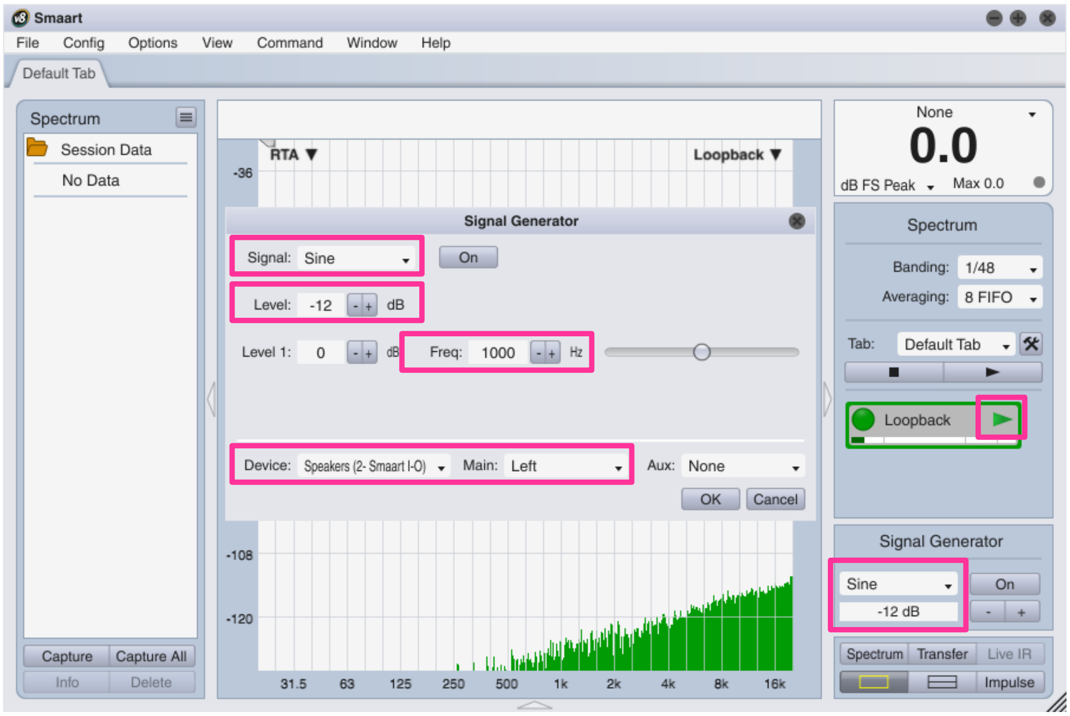


Figure 86: Signal Generator setup for distortion and overload example application

Make sure the spectrum measurement that you set up is still running, then adjust the signal level so that it is in the yellow portion of the measurement's input level meter. You can use the -/+ buttons on the signal generator or your external gain stage if applicable. If all goes well, you should see a nice clean spike on the RTA display at 1000 Hz like the one in Figure 87a. If you don't see anything, click anywhere in the left margins of the RTA graph to reset it to the default *y* range.

If you can't see the noise floor of your I-O device on the RTA graph, use the up/down arrow keys on your keyboard to slide the range of the graph up and down and the plus/minus keys to scale the *y* range until you can see everything from the noise floor up to 0 dB.

Now increase the signal level a little bit at a time until input starts to clip. As the signal level meter for the measurement starts to max out in the red zone, you should start to see additional spikes rising up on the RTA graph at integer multiples of 1 kHz as in Figure 87b. . Those are harmonic distortion products forming as overloaded input clips the signal and our nice clean sinewave starts to resemble something more like a square wave.

1. Undistorted sinewave on a log frequency scale
2. Clipped sinewave and distortion products on a log frequency scale
3. Clipped sinewave and distortion products on a linear frequency scale

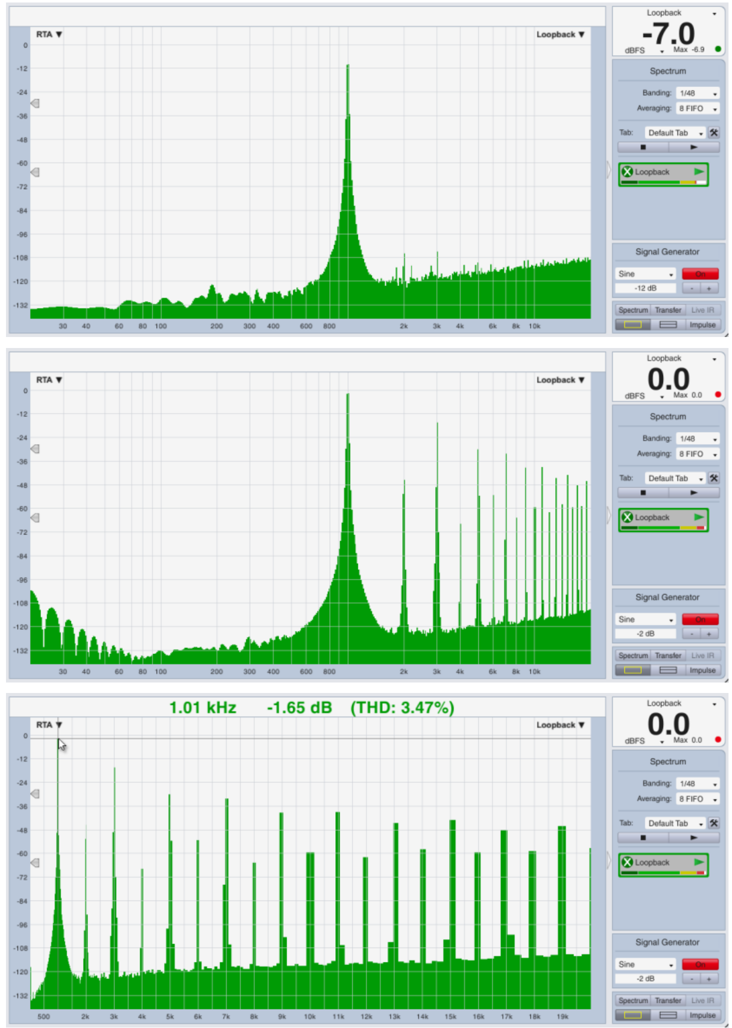


Figure 87: Measurement results for distortion and overload application example

The relationship between the harmonic frequencies becomes even more apparent if you switch the frequency scale of the graph to linear, so click on the word *Spectrum* at the top of the Control Bar or select *Spectrum* from the *Options* menu to open spectrum options. In the *Graph* Settings section of Spectrum options, change the *Frequency Scale* setting to *Lin* and click the check box labeled *Show THD* in the *RTA Display Settings* section. Click the *OK* button in the lower right corner of the dialog window to apply changes and exit the dialog. You should now see the distortion products from the overloaded input spaced at even intervals along the frequency axis of the plot. Also, if you put your mouse cursor on the peak at 1 kHz, Smaart will calculate THD and display the value in the cursor readout.

Be sure to go back in and change the frequency scaling for the RTA back to one of the logarithmic options (Decade, Octave or 1/3 Octave) before moving on to the next exercise. Linear frequency scaling is great for looking at harmonics and comb filters, but generally not that great for most other things that we do with Smaart.

Feedback Frequency Identification

For this next example, we need an actual sound system. It does not have to be a very elaborate sound system, just a vocal microphone and a mixer driving an amplifier and loudspeaker or a powered speaker will do the trick. The vocal microphone is routed through the mixer to the amp and loudspeaker and we acquire the output signal of the mixer as our measurement signal.

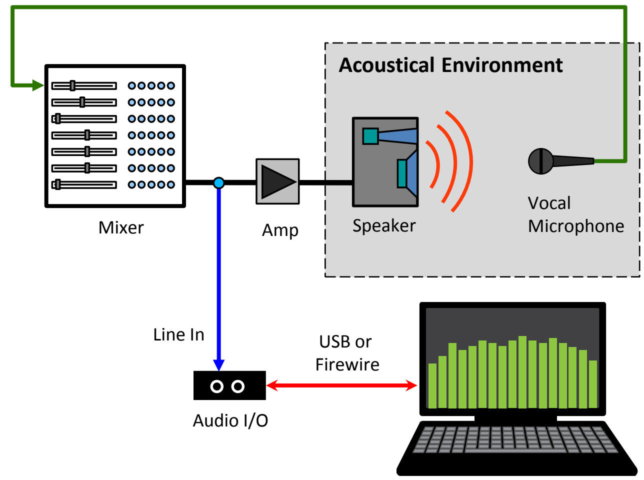


Figure 88: Measurement system setup for feedback study

In Smaart, set up a spectrum measurement for the input on the Audio I-O device that you are connecting to the mixer. See the *Distortion and Overload* example application on page 101 for instructions on how to do this if you need help. Click the *Spectrum* button in the Control Bar, and then click the button with the split rectangle graphic in the next row down. Your graph area should then look like Figure 89, with an RTA graph on top and a Spectrograph below. Set the *Banding* selector in the Control Bar to the right of the graph to 1/24-octave and *Averaging* to 1 Sec.

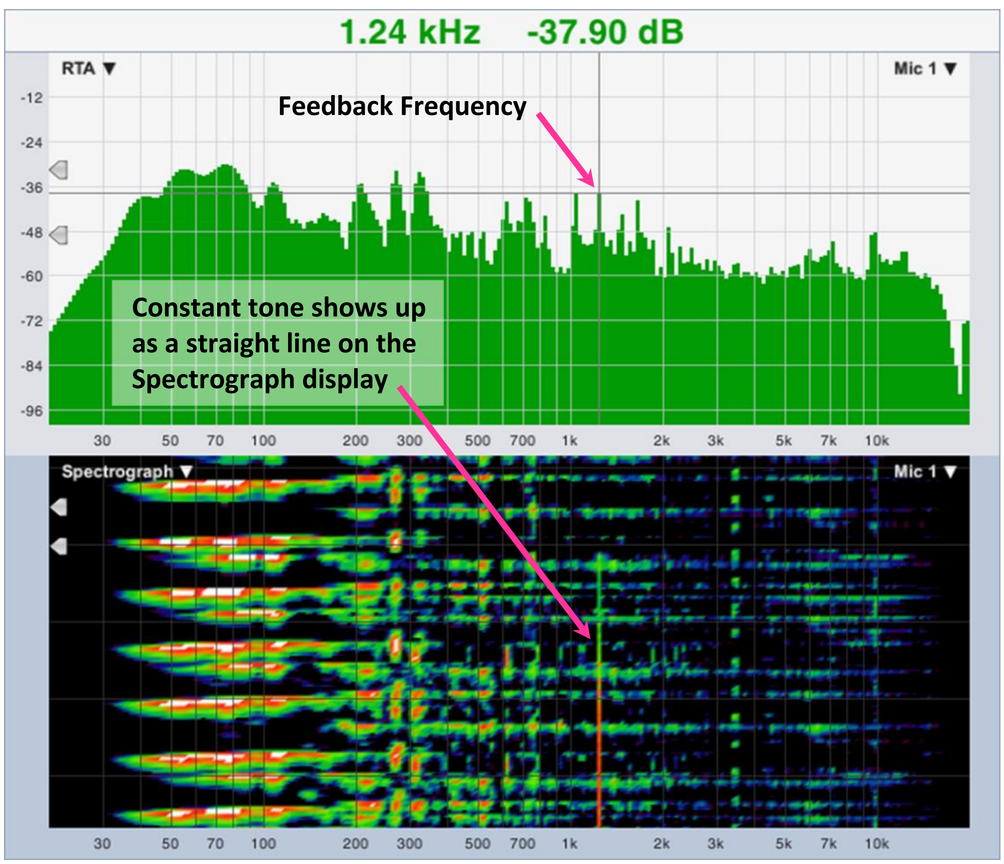


Figure 89: Feedback Study

Click the run button on your live measurement to start acquiring data and then slowly bring the microphone gain up until the system begins to feed back. You should see a vertical streak forming on the Spectrograph at the feedback frequency, like the one at 1.24 kHz in Figure 89. If you position your mouse cursor over the streak on the graph, you can read off the precise frequency in the cursor readout at the top of the graph area. For extra credit, plug a music player into the mixer and repeat the experiment with music playing. You should still find it pretty easy to identify the feedback frequency on the Spectrograph, whereas it becomes much harder to find on the RTA graph in the presence of a complex dynamic signal like music.

The spectrograph is also a powerful tool for helping to track down and identify audible problems other than feedback during performances. It is a common practice for mix engineers to route the solo bus output of a mixing console to an input for Smaart. Monitor engineers find this particularly useful for analyzing the spectral content of their mixes, or in general, for looking at the spectrum of any input signal before it is amplified by the sound system. FOH engineers will often have a microphone set up as well to monitor the acoustical output of the system in real time. Even the most trained ears can benefit from this. For example, if you are hearing a low mid buildup between 160Hz and 220Hz, the spectrum analyzer can help you see exactly what frequency is the main offender, and how much attenuation is needed to get it back in line.

Examining Interaction Patterns with the Spectrograph

The following is a simple technique that uses the Spectrograph for examining coverage and interaction patterns in loudspeaker systems. Simply put, you excite the system with pink noise – which should produce a relatively constant level/color at all frequencies on a spectrograph plot – and then move the measurement mic through the listening environment. Level variations from interactions, like the audible comb filtering caused by reflections, can be seen as interaction patterns on the spectrograph plot of the mic signal. Adjusting the dynamic range helps to better highlight the interaction patterns.

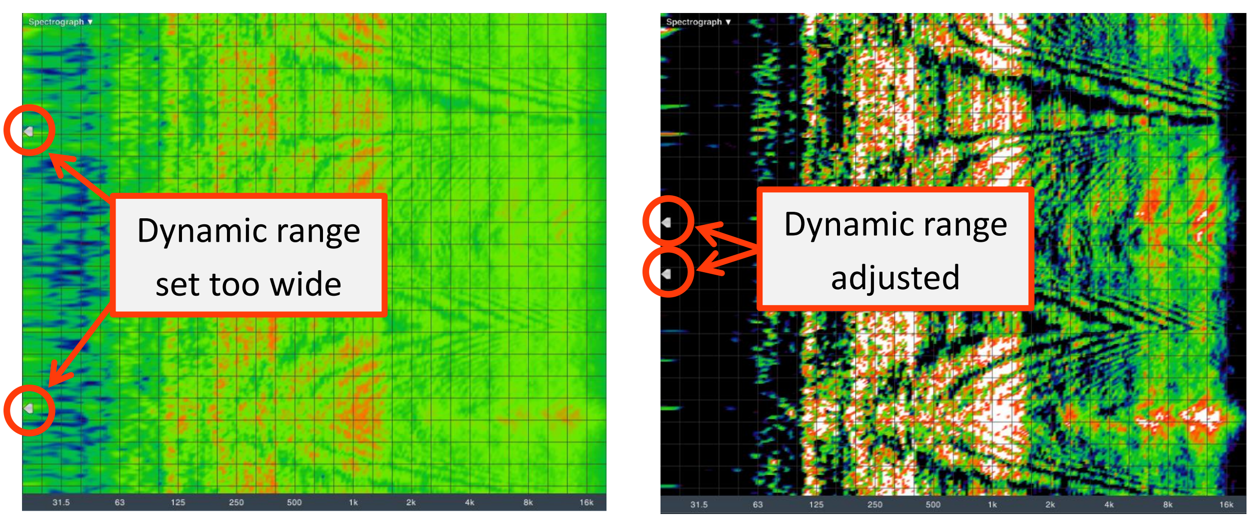


Figure 90: Spectrograph plot of comb filter interaction patterns

Chapter 6: Dual Channel Transfer Function Measurements

The transfer function is a dual-channel measurement technique that determines a system's frequency response by comparing its input signal (the reference signal) to its output (measurement signal). The result of this measurement is a complex signal that represents the difference between the measurement and reference signals in both magnitude and phase. The measurement results show us the aggregate processing behavior of the system under test (SUT) as a function of frequency or time.

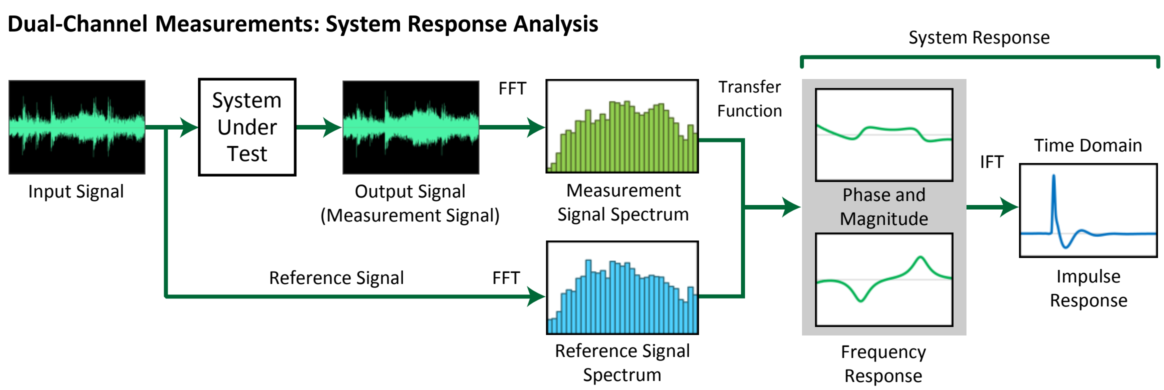


Figure 91: Block diagram of a transfer function or dual-channel impulse response measurement

The word "system," in this case means everything that affects the spectral energy content and timing of the reference signal, from the point where it was introduced into the signal chain, all the way through to the point where we picked up the resulting output as our measurement signal. In the case of acoustical measurements (captured by means of a microphone), the definition of SUT therefore includes the acoustical path from the loudspeaker as well as the electronic path and the loudspeaker system. If we were measuring just a single processor channel from its input to its output, then that one channel is the SUT – for electronic measurements of a single device we might say "device under test" (DUT) rather than SUT but either term is technically correct in that case.

The transfer function allows you to examine the frequency response of components of a sound system, both electrical (EQ's, mixers, processors) and electro-acoustical (loudspeakers, their drive electronics and their acoustical environment). This type of measurement is extremely useful in a wide range of applications, including loudspeaker design, equipment evaluation, equalization and sound system optimization.

Data from transfer function measurements is presented in three different forms in Smaart, on three chart types: magnitude response, phase response, and live impulse response (Live IR). The first two (*Magnitude* and *Phase*) are frequency domain charts with frequency on the horizontal *x* axis and the dependent variable (magnitude or phase) on their vertical *y* axis. The *Live IR* is a time domain chart with *time* on the *x* axis and either linear amplitude or decibel magnitude on its *y* axis. A related measurement called coherence is also calculated from the same data. Coherence is displayed on the *Magnitude* graph as an indicator of the quality of the data transfer function.

Two groups of settings determine the appearance and behavior of these displays:

- Measurement settings affect how data is acquired. These are set from the *Measurement Config* page of the *Configurator* dialog (*Config* menu > *Measurement Config*) and we have discussed them in some detail in Chapter 3 (see page 57). In this section, we will look more specifically at how some of those options directly affect the *Magnitude* and *Phase* and *Live IR* displays.
- Display settings affect how transfer function measurement data is displayed after it is acquired but do not change the underlying measurement data. These options primarily reside on the *Transfer Function* page of the options dialog (*Options* menu > *Transfer Function*), which we will be looking at in this section.

As was the case with Spectrum measurements, the line between the measurement and display functions gets a little blurry in places. Fractional octave smoothing, for example, is technically a display function that does not affect the underlying data, but we group it with the measurement parameters for practical reasons. Thresholds for magnitude and coherence can be set from either place. The basic organizational intent however, is that display settings mainly reside in *Transfer Function* options, and measurement parameters are located in *Measurement Config*.

Dual Channel Measurement and Display Configuration

Transfer Function Control Bar

At a glance, the Control Bar for transfer function displays looks a lot like the one for Spectrum measurements. Obviously it says "*Transfer Function*" at the top instead of "*Spectrum*." If you hover over the heading with your mouse cursor, it turns into a button that opens *Transfer Function* options. Here again, we have an *Averaging* selector for the active measurement but instead of *Banding,* there are separate smoothing controls for the magnitude and phase displays (*Phase Smooth* and *Mag Smooth*)

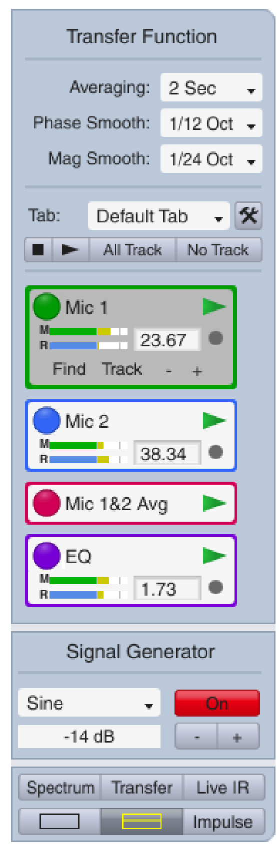


Figure 92: Transfer Function Control Bar

At the measurement configuration level, all three of these settings can be set globally, for all transfer function measurements or specifically for individual measurements. If the current active measurement uses global settings for any of these parameters (as is most commonly the case) then the controls on the Control Bar will apply to the global settings and changes made to these controls will flow through to any other measurements that subscribe to the global settings. For any that are set locally, at the measurement level for the active measurement, the corresponding control on the Control Bar affects only the active measurement.

Another difference between the *Spectrum* and *Transfer Function* Control Bars are their live measurement controls. The *Tab* selector works identically to the one for spectrum measurements but in addition to the Stop All (■) and Run All (►) buttons, we also have *All Track* and *No Track* buttons that turn delay tracking on and off for all measurements in the group.

Individual measurement control blocks for dual-channel transfer function measurements include the standard color-tile/hide button, measurement name and run/stop button found in other live measurement controls. In addition, each control block has a delay field and two input level meters; one for the measurement signal (labeled "M") and another for the reference signal (labeled "R"). The control block for the *active* transfer function measurement trace expands to include an extra row of hover buttons below the signal level meters. These include a *Find* button that invokes Smaart's *Delay Finder*, a Track button that toggles delay tracking on or off, and -/+ buttons that nudge the delay time setting up or down by one sample. In Figure 92, the measurement block labeled *"Mic 1"* is active.

The delay field in each measurement block is directly editable. You can click it with your mouse to select it, then type in a new value and press the [Enter] key on your keyboard to set the change. The gray circle to the right of the delay field turns yellow when delay tracking is active and works as a button to turn tracking on and off when you click it with your mouse.

Transfer Function Measurement Configuration

If you click the little hammer and wrench button on the Control Bar (see Figure 92) to *open Measurement Config,* then select a transfer function measurement in the tree view or double click the name of one on the measurements table in a tab view, you see that there are several more measurement parameters than there were for spectrum measurements and more of them are localizable.

Figure 93: Transfer function measurement parameters

FFT Size

The FFT selection for transfer function measurement can be set globally or for individual measurements, and not just globally for all measurements. The *FFT* selectors for transfer function measurements includes the same selection of power-of-two FFTs offered for spectrum measurements, plus one additional option called *MTW*.

MTW stands for multi-time-window. This is the default FFT selection for transfer function measurements and for a large majority of system tuning applications, there may rarely be any real need to change it. Rather than taking a single FFT for each input signal (reference and measurement) at a single sample rate, MTW uses a series of sample rate decimations and varying FFT sizes to produce a measurement with different time and frequency resolutions in different frequency ranges. There are a several benefits associated with this approach.

One benefit of MTW is that it sidesteps some of the time vs. frequency resolution trade-offs inherent in FFT analysis. Having only about 840 frequency data points makes MTW measurements much easier to read than a single-size FFT measure with comparable low-frequency resolution. Another is that the use of shorter time windows at higher frequencies makes the coherence function a much more useful tool for detecting timing mismatches between the reference and measurement than any single-size FFT based measurement with comparable low-frequency resolution. At 48k sampling rate, MTW produces better than 1 Hz resolution at the lowest frequencies, compared to ~1.5 Hz for a 32K FFT, with higher computational efficiency, and with nearly 20 times fewer frequency data points than a 32K FFT, it is much less work for your graphics hardware to plot.

Magnitude Averaging Type (Mag Avg Type): Polar vs Complex

Magnitude Averaging Type is a global parameter for transfer function measurements that applies to all 2-channel measurements. There are two options: *Polar* or *Complex*. The difference between them under the hood (internally) is that Complex averaging maintains separate running averages for real and imaginary data and calculates magnitudes for display from the averaged complex data. Polar averaging calculates magnitude for each incoming measurement update and maintains a single running average of the magnitude values.

In practical terms, the main difference between them is that magnitude (Polar) averaging lets in more reverberant energy, which may tend to agree better with what you hear, particularly for musical program material. Polar averaging can also be more stable than complex averaging under difficult measurement conditions, where there is a lot of background noise and/or wind or physical movement. Complex averaging rejects more reverberant energy as noise than polar averaging and could offer better noise immunity as well. Because human hearing is quite sophisticated when it comes to processing sounds in reverberant environments, complex magnitude averaging may give you better clues regarding speech intelligibility than polar averaging.

Note that these two averaging options apply only to how magnitude data is processed. Phase averaging is based on complex averaged data in both cases.

Smoothing

Smoothing helps to reduce visual noise and ripple in transfer function measurement data by averaging each frequency data point with some number of the points on either side. A center weighted averaging window is used that expands logarithmically, ascending in frequency, similar to how fractional octave banding works. Choices for smoothing are in fractional octave increments – larger fractions mean more smoothing. There are separate smoothing controls for phase and magnitude (Mag) data because we tend to look for different things on the two displays. It is common to use more smoothing for phase data, where you tend to be most interested in overall trends than for magnitude data, where you might wish to see more detail.

Weighting

Weighting applies a weighting curve to transfer function measurements, either locally or globally. Common weighting curves include *A* and *C* weighting used for SPL and Leq measurements and the *X* curve used for cinema sound systems.

Delay Compensation

Delay compensation is a crucial factor in transfer function measurement. The reference signal for the measurement, regardless of source, is typically a direct connection to our measurement system, meaning it travels through a piece of wire to the input of the measurement system at more or less the speed of light. The measurement signal, being the output of the system under test, is subject to delay from filtering, throughput delay in digital devices, intentional alignment delay, and of course in the case of acoustical measurements, propagation delay due to traveling through air at the speed of sound.

All of this can introduce tens of milliseconds of time offset between the reference and measurement signals and we must compensate for that offset by delaying the reference signal to match the arrival time of the measurement signal. Every two-channel transfer function measurement has a built-in delay line for just that purpose. In the *Measurement Config* dialog, delay times for each live transfer function measurement appear in the table on the *Group* tab and on the individual measurement settings tabs. If you happen to know what the delay time should be for a given measurement you can enter it manually in any of these locations. Otherwise, we need to measure it.

Measuring Delays

There are multiple methods for finding relative delay time between two signals in Smaart. Most commonly you use one of two automated routines (delay tracking or the *Delay Finder*) that are based on impulse response measurements. In trickier cases, you can switch to IR mode, measure the impulse response and visually analyze the results. Experienced users also use the *Phase* display to fine tune delay times or deal with difficult cases such as subwoofer measurements in a noisy environment.

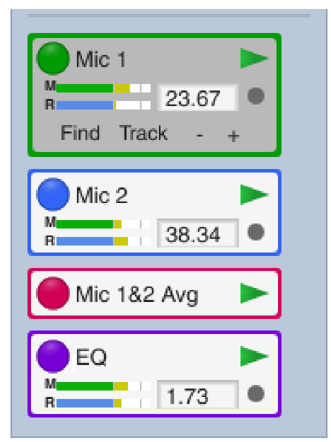


Figure 94: Delay Finder button for the active transfer function measurement

The Delay Finder

Smaart's Delay Finder is an automated routine for finding the delay time for the active transfer function measurement. It works by measuring the impulse response of the SUT and then scanning the IR it to find the highest peak, which will normally represent the initial arrival of the measurement signal at the measurement point. In acoustic measurements, we call this the first arrival or arrival of direct sound.

To run the *Delay Finder*, make sure the active graph is a transfer function *Magnitude*, *Phase* or *Live IR* display; so that the transfer function measurement controls are visible in the Control Bar to the right of the graph area. Make the measurement whose delay time you want to find the active measurement, by clicking on its control block in the lower portion of the Control Bar – the measurement must be running in order to select it as the active measurement. Make sure that the signal source being used to excite the SUT is turned on and the input levels for the measurement are running at reasonable levels, and then click the *Find* button that appears below the meters in the active measurement control block. The *Delay Finder* window appears; Smaart runs the measurement procedure and reports its result when finished.

If the measured delay time seems reasonable and you are happy with the results, click the *Insert* button to assign the measured delay time to the current measurement and exit the Delay Finder dialog. If not, you can click the Find Delay button to run the measurement again, or click *Cancel* to exit the dialog without assigning the measured delay time.

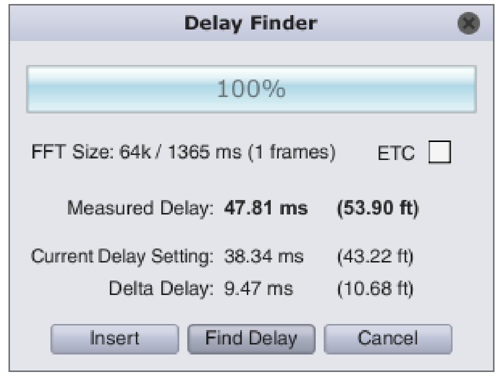


Figure 95: The automated delay finder

The FFT size and number of averages for the *Delay Finder* are set from the *Delay* tab of the main Options dialog (*Options* menu > *Delay*). The default is a 64K FFT with no averaging, which works out to a time constant of 1365 ms at 48k sampling rate. This is sufficient for finding delay times at distances up to a about 450 hundred feet (140 meters) from a source – a good rule of thumb is that the FFT time constant should be least 3x greater than the expected delay time. When measuring from extremely long distances or using a sample rate greater than 48k, you may need to increase the FFT size. When working in very noisy surroundings, increasing the number of averages may help as well.

Notice that when you run the *Delay Locator*, Smaart automatically calculates the difference (*Delta Delay*) between the measured delay time and the current delay setting for the measurement. This is a handy tool for finding relative delay time between two loudspeaker systems, for example a main PA system and a delay speaker. The procedure is to measure the first (later arriving) system and set your delay time, then mute the first system, turn on the second system and run the delay locator again.

Selecting the *ETC* check box in the *Delay Finder* dialog tells Smaart to use the Envelope Time Curve (ETC) of the impulse response, rather than the IR itself to find the delay time. This will often result in a slightly different delay time. It is possible that one may provide a better answer than the other.

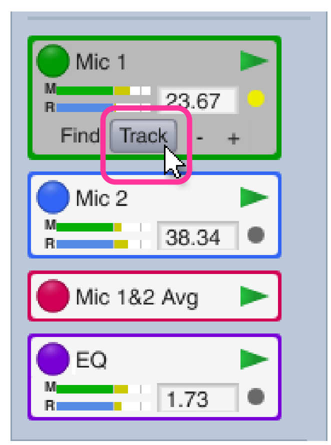


Figure 96: Delay Tracking button for the active transfer function measurement

Delay Tracking

Smaart's delay tracking feature is designed to keep transfer function measurements aligned in situations where the delay time may change from one measurement update to the next, for example, when measuring in windy conditions or while the microphone is being moved to a new location. You can also use it as a quick and dirty delay locator when measuring relatively short delay times under well-behaved measurement conditions. Just turn tracking on and let it find the delay time and lock on. This can work for delays of up to 80-90 milliseconds in electronic measurements. For acoustical measurements made in the presence of reverberation and noise, the effective limit may be more like 50-60. Bear in mind that delay tracking consumes computing resources, so you generally want to keep it turned off when you do not expect the delay to change while the measurement is running.

Magnitude Response

The transfer function Magnitude graph shows both the magnitude portion of the frequency response of the system under test (SUT), and Coherence for the active transfer function measurement. The magnitude plot shows relative gain and attenuation in the output of the system at each frequency.

If the reference and measurement signals are identical in level at all frequencies, the magnitude trace is a flat line at 0 dB. If there is an overall level difference between the two signals, the centerline of the measurement moves up or down on the graph – up means the measurement signal is coming in at a higher level, relative to the reference signal, down means the reverse is true. If the SUT produces a relative gain at some frequencies and relative attenuation at others (as is usually the case with real-world sound systems), the magnitude trace will deviate above the centerline of the measurement at frequencies where there is a relative gain and dip below it in regions of attenuation.

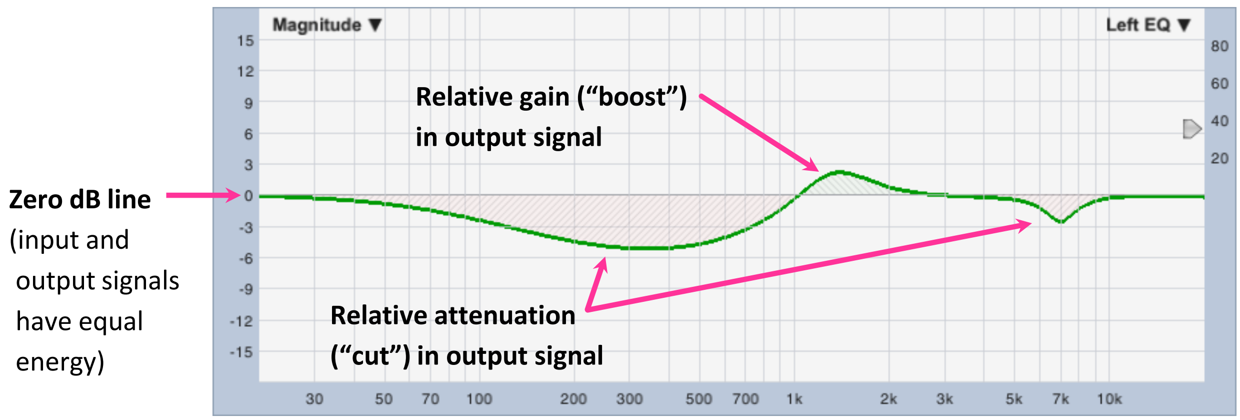


Figure 97: The transfer function Magnitude graph

Because we are directly comparing the signal going into the SUT to the output signal that the system produces in response to it, you get the same response curve using virtually any broadband signal with sufficient energy at all frequencies to resolve the measurement, including music. Unlike spectrum measurements, the shape of the response curve is not dependent on the spectrum of the input signal. The resulting magnitude response ends up looking very similar to a fractional octave spectrum measurement made using pink noise, particularly when *Polar* averaging is used. When *Complex* magnitude averaging is selected, RTA and transfer function magnitude measurements of the same system *can* look significantly different in some cases, owing to the tendency for complex averaging to reject reverberation as "noise."

When running multiple live transfer function measurements it is a good practice to match the sensitivity of each measurement so that that their overall magnitude levels match. This makes the overall levels of each measurement directly comparable to each other and directly relatable to relative sound pressure level. A straightforward method for sensitivity matching is to measure the same source from exactly the same position with each microphone, adjust their levels to match in that position and then don't touch the preamp settings afterward. Of course, no two microphones have identical frequency response but any mic that bills itself as a measurement microphone should have very nearly flat response up to 10-12 kHz at least, so that is generally the range that you want to concentrate on most.

Note that sensitivity matching is not the same thing as sound level calibration. You could however, accomplish the more or less the same result using a sound level calibrator, by adjusting the input gains for each microphone to get the same full-scale amplitude for each. In that case, the calibration offset for SPL measurement would end up being the same for each microphone.

Phase Response

The transfer function *Phase* graph shows the phase portion of the frequency response of the system under test (SUT). Phase is plotted with frequency in Hertz on the *x* axis and phase in degrees on the *y* axis. Phase, or phase shift is a measure of the relative time relationship between two signals as a function of frequency, expressed in terms of cycle time. Like the *Magnitude* plot, the slope of the phase trace is a flat when the reference and measurement signals for the transfer function are identical and arrive at exactly the same time. *Unlike* the magnitude trace, the phase trace does not go to 0° at frequencies where the two signals are arriving at the same time; it just flattens out.

With just a few real exceptions, the key to reading the phase trace is to almost ignore the numbers on left the side of the graph and pay attention only to the slope of the line. When the line slopes upward, the measurement signal is arriving before (leading) the reference signal. When it flattens out and trends sideways, the two signals are arriving at the same time. When the line slopes downward, the measurement signal is lagging behind the reference signal. If you simply repeat those three things to yourself until they are burned into your brain, you will immediately know more about reading a phase trace than nearly everyone you meet.

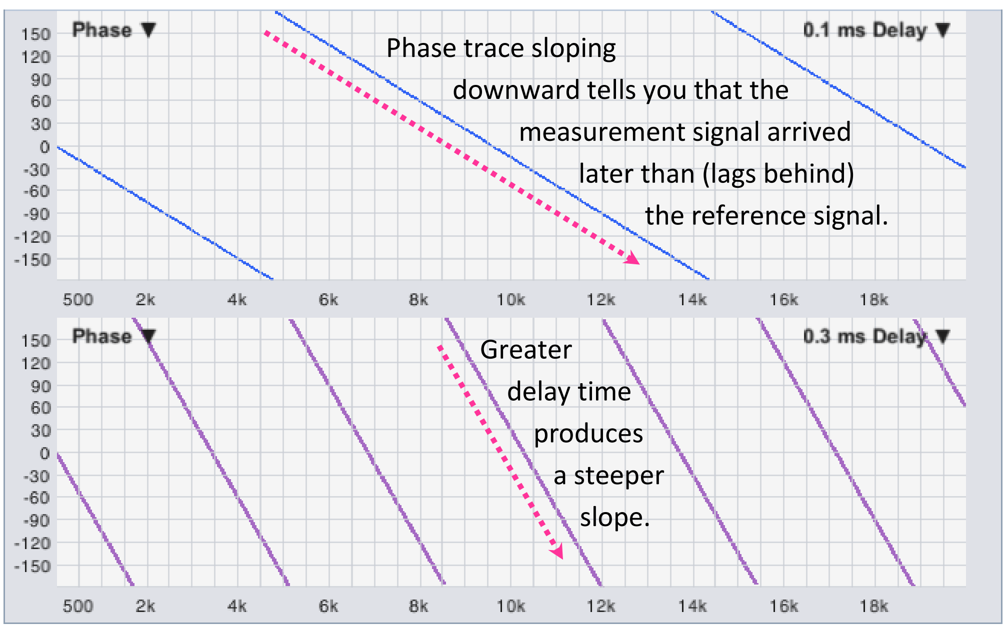


Figure 98: Uniform (linear) delay on a linear frequency scale

When there is a uniform time offset between two otherwise identical signals, you will see that the phase trace is a straight line on a linear frequency scale, sloping upward or downward at a constant rate of change; the greater the delay time, the steeper the slope. On a logarithmic frequency scale, the straight line becomes a curve and the wraps become more tightly packed as you ascend in frequency, but the information is the same. The reason why the phase trace "wraps" – that is, it runs off the top or bottom of the graph periodically and then reappears on the opposite side – is that we are measuring time based on the cycle times of sinewaves, and everything that we actually *know* about the timing relationships between the two signals is confined within a 360° range.

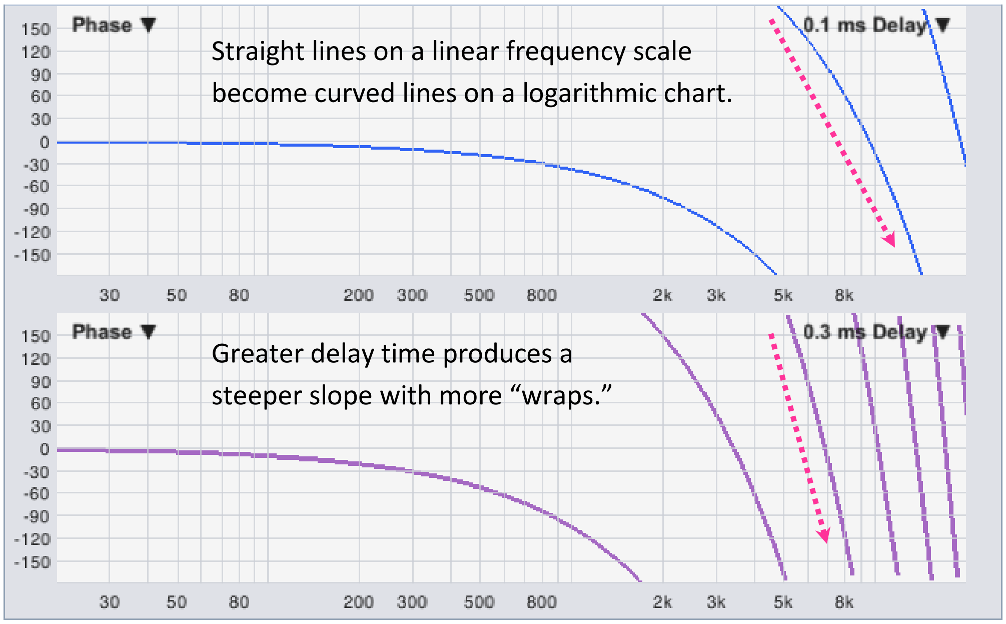


Figure 99: Uniform (linear) delay on a logarithmic frequency scale

For example, a 10 Hz sinewave has a cycle time of 0.1 seconds, or 100 milliseconds (1÷*f*, where *f* is frequency), meaning that two identical signals that are offset in time by 25 ms, are offset in phase by one quarter of a cycle, or 90 degrees at 10 Hz. At 20 Hz, the same 25 ms of delay represents one-half cycle or 180° degrees of phase shift. It follows that 40 Hz sinewave cycles a full 360° in 25 milliseconds, but here is the tricky part: so does an 80 Hz sinewave, and a 160 Hz sinewave, and a 320 Hz sinewave… because our phase "clock" only goes up to 360°. It resets to zero every time a sinewave completes one full cycle. You cannot tell time beyond 1÷*f* seconds by looking at phase at a single frequency. It is only when you put multiple data points together that you can begin to see phase relationships in the context of a larger timeframe.

You could think of the standard ("wrapped") phase display as a continuous line drawn on a paper tube, which we slice along its length and lay out flat so that we can read it. Anywhere the line crosses the point where we sliced the tube, the trace jumps from the top of the graph to the bottom, or vice versa. If you use the up/down arrow keys when a phase graph is selected as the active graph in Smaart or click on a phase trace and drag it up or down to change the range of the graph, it is analogous to gluing the tube back together and then slicing it again at a different point. This, incidentally, is also why Smaart does not draw vertical line segments between wrap points, as you often see on phase graphs. Connecting lines at the wrap points do not represent actual data and properly should not be there.

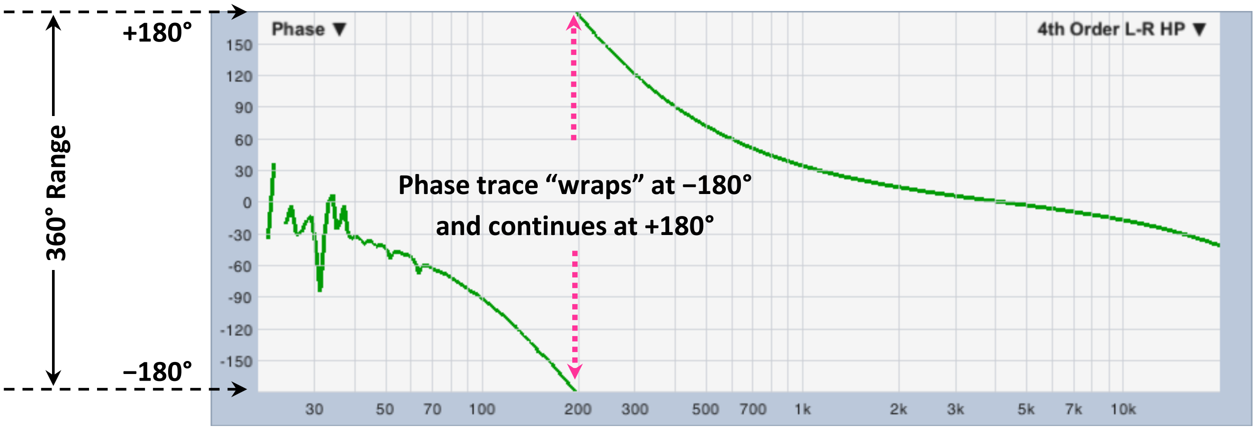


Figure 100: The standard "wrapped" Phase display

Up to now, we have been talking about phase shift in terms of two otherwise identical signals arriving at two different times, but of course, most of the things that we measure with Smaart do more to an input signal than simply delaying it. When you put a signal into a system under test, the signal that comes out typically has more energy than the input signal at some frequencies and less at others. ***This is called filtering.*** You can actually think of transducers, loudspeaker systems, and even entire sound systems as bandpass filters. All of the above allow energy that falls within some frequency range to pass through relatively unmolested, while energy outside that range is significantly attenuated – this is to say, that they all have a defined passband with transition bands and stopbands on either side, which is pretty much the functional definition of a bandpass filter.

In physical systems, any process that affects the spectral content of a signal also affects its timing. You cannot change magnitude response without affecting phase response. Those are the rules.

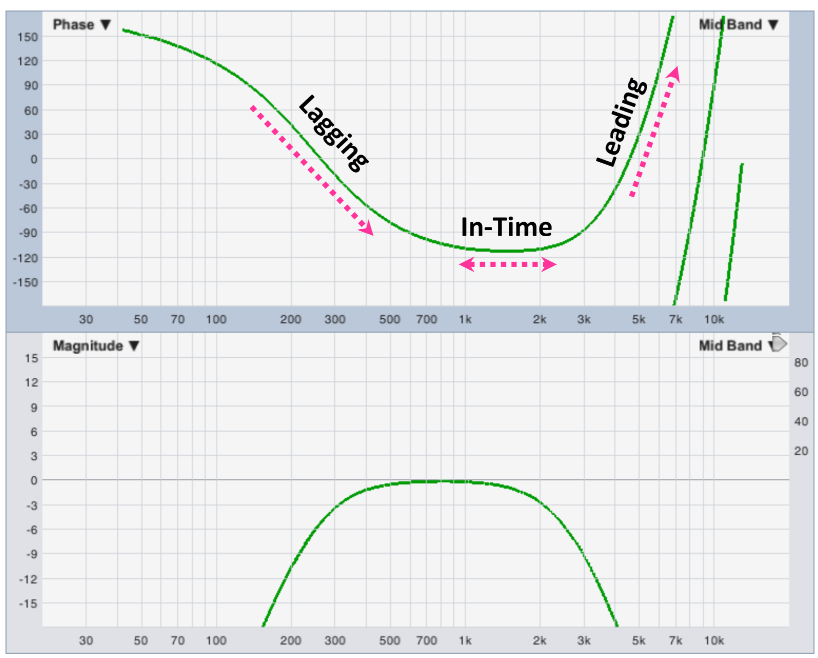


Figure 101: Phase and magnitude response of a 4th order Linkwitz-Riley bandpass filter

Now at this point, some alert readers are no doubt thinking, "Hey, wait a minute. What about linear-phase FIR filters?" Well, in fact, those *do* produce phase shift, however they are designed with a symmetrical impulse response that induces exactly the same amount of phase shift forward and backward, so that phase shift in the back half of the filter exactly cancels out the phase shift from the front half. The price that you pay is an overall delay time at all frequencies equal to half the length of the filter kernel, which is why they are called linear-phase filters, and not zero-phase-shift filters. That delay time penalty also tends to limit the usefulness of FIR filters in live sound reinforcement applications, particularly at lower frequencies.

Symmetrical FIR filters aside, infinite impulse response (IIR) digital filters, analog filters, and other continuous-time processes that affect the frequency content of signals, such as air loss and acoustical reflections, all produce asymmetrical impulse responses. They affect relative timing differently at different frequencies. The phase response of a bandpass filters typically leads at some frequencies, lags at others and is in-time at others still, so when you need to align two bandpass filters in time – whether it's two drivers, two cabinets or two entire subsystems – there isn't any single right answer that works at all frequencies. You have to choose a frequency range where you want the two functions to align. This is where the phase display comes in handy.

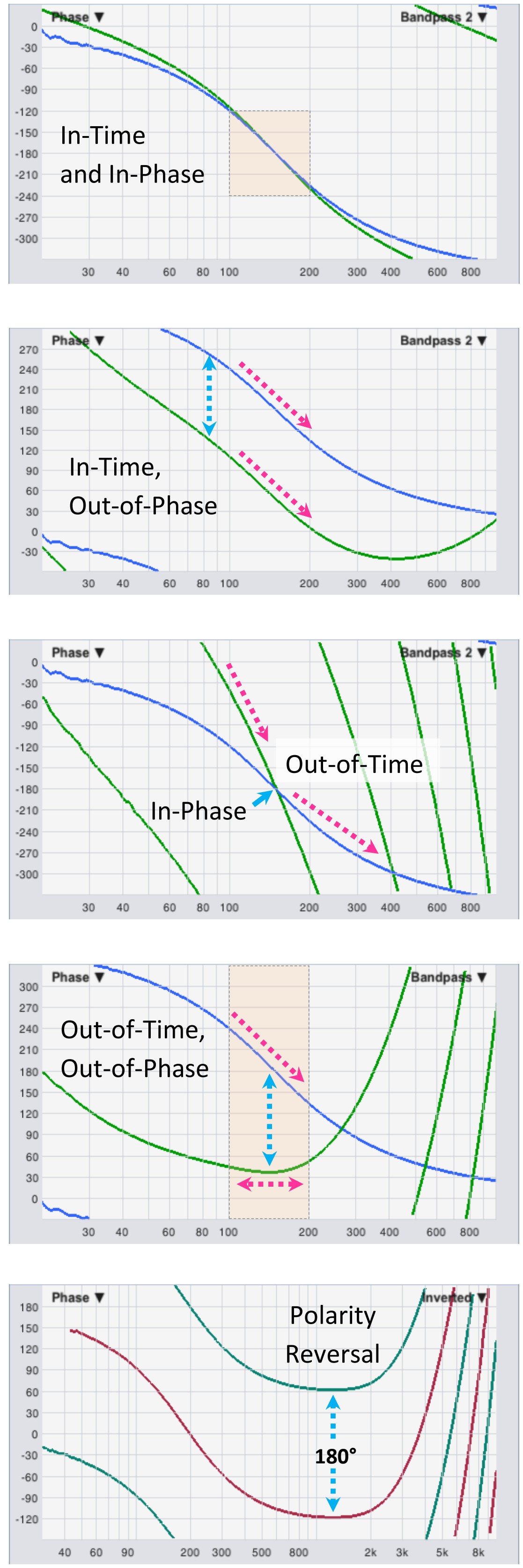


Figure 102: Comparing phase traces

Comparing Phase Traces

When comparing phase traces, remember that slope tells you about arrival time (the up/down/sideways rule) and vertical position on the graph shows phase shift. With those two thoughts in mind:

- At any frequencies where two traces have the same slope, they are aligned in time – that is, both measurements are showing the same relative delay time at those frequencies, regardless of their relative positions on vertical axis of the graph. We refer to this as being "*in-time*."
- Whenever two phase traces have the same slope and lay right over each other on the graph, they are both *in-time* and *in-phase* with one another.
- When two traces cross each other on the graph but have different slopes, they are *in-phase* at their crossing point, but *out-of-time* with each other (arriving at different times).

- Two traces with different slopes and different vertical positions on the graph at some frequency range of interest are *out-of-time* and *out-of-phase* with each other.
- If you see two phase traces that look identical or nearly so, but they are separated vertically on the graph by exactly 180°, that tells you there is a relative *polarity reversal* between them. (This is the main exception to that rule about looking at the slopes and ignoring the numbers that we mentioned earlier.)

Unwrapping the Phase Display

As we have discussed in the last few pages, everything that we actually know about phase shift lies within a 360° range that we typically plot as ±180°. Earlier, we used the example of a 25-millisecond delay, which produces the same 360° of phase shift at 40 Hz, 80 Hz, 160 Hz, 320 Hz… all the way up to the Nyquist frequency for our sampling rate, where we finally run out of frequencies. Of course, *we* know that each time the frequency doubles, twice as many cycles fit into the same time span, but phase does not know that. You could think of it as an analog clock with no hour hand. The minute hand can tell you how much of the current hour has elapsed but cannot tell you what hour of day or night it is.

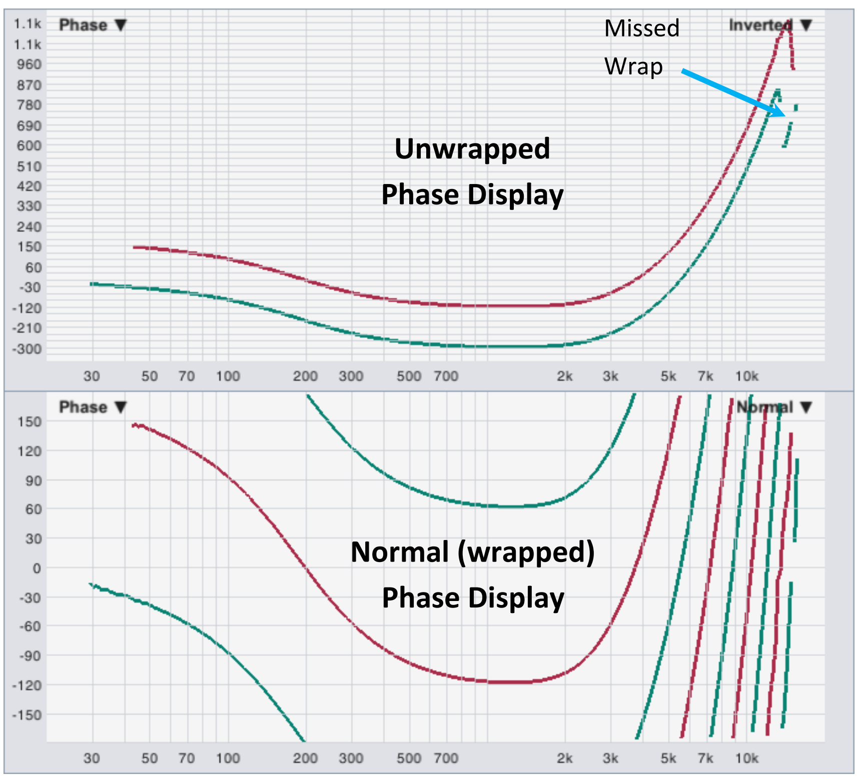


Figure 103: Phase on an unwrapped, versus wrapped (normal) phase display

We can however, *infer* things from phase relationships that the phase values can't tell us by themselves. To extend the clock analogy, even though a clock without an hour hand cannot tell us what time it is over any timeframe longer than one hour, we *could* keep track of how many hours have elapsed since we started watching it by counting of the number of times the minute makes one complete revolution around the dial.

This is essentially how an unwrapped phase display is constructed; by counting the wrap points and adding or subtracting from the count each time a wrap occurs. The procedure becomes a bit more complicated by the fact that our "clock" in this case can run both forward and backward, and may occasionally start spitting out random numbers in noisy areas of the measurement, such as the nulls of comb filters and at frequencies outside the passband of the SUT. Reverberation can be an issue as well. All this is to say that unwrapping phase can be a little iffy, and it definitely works better in some cases than in others. However, an unwrapped phase display can still be useful when it works.

To display unwrapped phase in Smaart, click on the *Transfer Function* label above the active measurement controls on the Control Bar or select *Transfer Function* from the *Options* menu to bring up the *Options* dialog window with the *Transfer Function* page selected. In the *Phase* section of *Transfer Function* options, click on the *Unwrap Phase* check box to select it, then click the *OK* or *Apply* button at the bottom of the dialog window to apply the change. You can turn it off the same way.

Like the normal (wrapped) phase display, the unwrapped phase graph plots frequency in hertz on the *x* axis and phase in degrees on the *y* axis. Unlike the standard phase display, the vertical axis of the graph is not limited to a constant 360° range. The *Unwrapped Phase Range* settings in the *Phase* section determine the initial range of the unwrapped phase display. You can also zoom its vertical range using the [+]/[−] keys on your keyboard or right-click and drag with your mouse on the plot to select an *x*/*y* range for display, as you can most other plots in Smaart.

Phase as Group Delay

Another thing we can infer from phase relationships is delay time by frequency, or group delay. Since we know that the slope of the phase traces becomes steeper as delay time increases, we can use the rate of change between neighboring frequency points to estimate delay time by frequency. This can be a very useful measurement and because support for the function is very localized, a few bloopers here and there won't mess up the entire measurement, as can happen with the unwrapped phase display. You can still run into problems if there is significant ripple in the phase trace however, which may be an issue when measuring in a very reverberant environment.

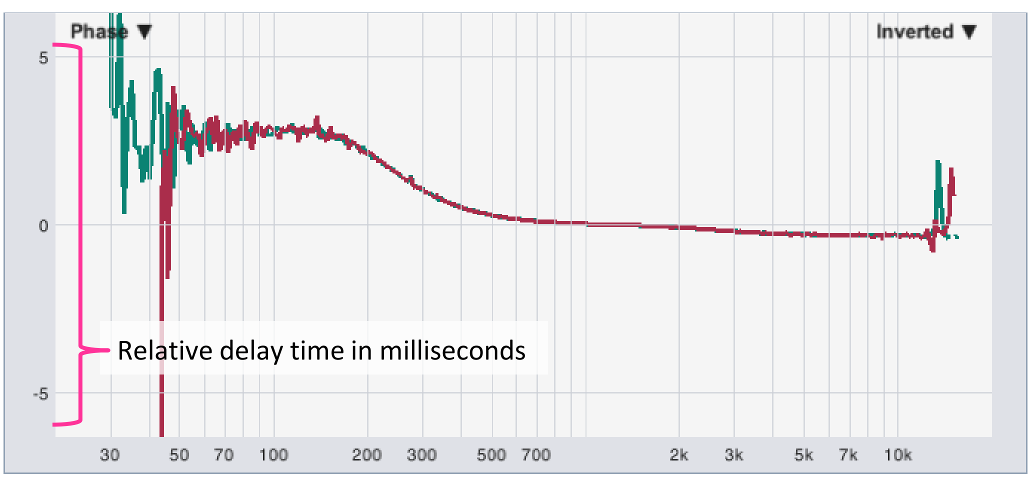


Figure 104: Phase shown as group delay

To plot phase as group delay, click on the *Transfer Function* label above the active measurement controls on the Control Bar or select *Transfer Function* from the *Options* menu to bring *Transfer Function* options, and then click on the *Phase as Group Delay* check box to select it. Clicking the *OK* or *Apply* button at the bottom of the dialog window applies the change. In group delay mode, the *Phase* display plots frequency in hertz on the *x* axis delay time in milliseconds on the *y* axis. If phase is linear, group delay will be constant and the group delay plot will be a flat on the graph at the delay time. If phase is non-linear, group delay will vary by frequency. To turn the feature off, open *Transfer Function* options and click its check box again to unselect it and then apply the change.

Coherence

Coherence is a statistical estimation of the causality or linearity between the reference and measurement signals in a transfer function measurement. Coherence does a good job of detecting contamination of the measurement signal by unrelated signals such as background noise and reverberation, and it is sensitive to timing mismatches as well. We use it in Smaart to gauge the quality of transfer function measurement data, frequency by frequency, in real time. Additionally, since the same factors that affect coherence (mainly noise and reverberation) also affect speech intelligibility, the coherence trace can also give you a sense of how intelligible a system is.

The coherence calculation essentially asks the question, “How confident can we be that what we are seeing in the measurement signal at this frequency was caused by the reference signal?” The answer is a number between zero and one, which Smaart displays as a percentage. A value of 100% indicates perfect correlation between the two signals and zero means there is no discernable relationship between them.

In nuts and bolts terms, coherence works by comparing the cross-spectrum (the frequency-domain representation of a cross-correlation) of the reference and measurement signals to the product of their averaged power spectra. That means it must be calculated across multiple readings of the two signals in order to be meaningful. If you looked at just a single reading of any pair of signals, coherence would always be 100% for all frequencies, and so the feature turns itself off when averaging is not in use.

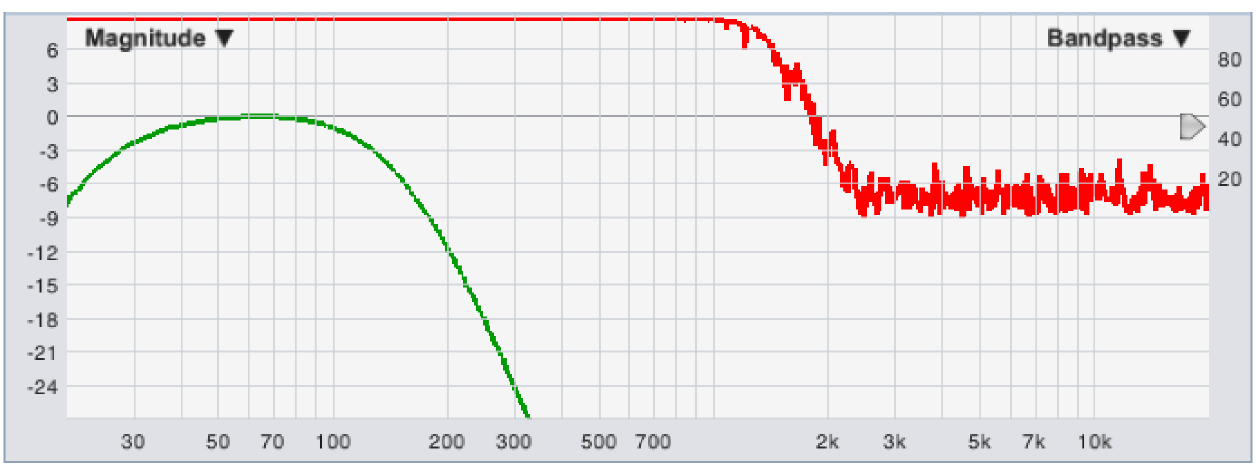


Figure 105: The coherence trace is plotted on scale of 0-100 in the upper half (or quarter) of the magnitude graph.

The Coherence Display

The coherence trace in Smaart is plotted in the upper portion of the Magnitude graph – either the top half or optionally, the topmost quarter – with frequency on the *x* axis of course and coherence, as a percentage value between zero and one hundred, on the *y* axis. Coherence is always calculated for all transfer function measurements that use averaging, but only the coherence trace for the topmost magnitude trace is plotted – in other words, the trace whose name appears in the upper right corner of the graph. This can be a live measurement or a stored trace, whichever is currently at the front on the *z* axis of the *Magnitude* graph.

The little arrowhead-shaped widget that you can see pointing at the coherence scale on the right edge of the *Magnitude* graph sets the threshold for the coherence blanking function. Coherence blanking removes questionable data from magnitude and phase traces at any frequency where coherence does not meet or exceed the specified threshold. You can click on the widget with your mouse and move it up and down to change the threshold. Coherence blanking works for all displayed traces on the *Magnitude* and *Phase* displays that use averaging, not just the front one, and it works even if the coherence trace is not displayed.

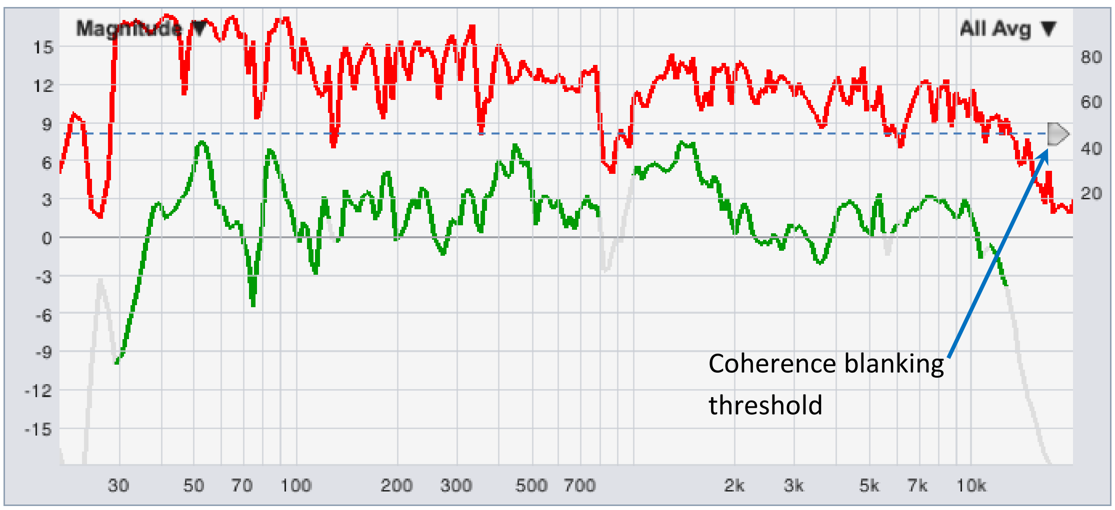


Figure 106: Coherence blanking

Display options for the coherence trace are found in the *Coherence* section of the *Transfer Function* options (*Options* menu > *Transfer Function*). These include *Show Coherence*, which turns on plotting of the coherence trace and *¼ Height*, which squeezes the coherence trace into to the top quarter of the *Magnitude* graph instead of the top half. The *Blanking Threshold %* field echoes the setting of the on-graph threshold widget and can be used to set the threshold to a specific numeric value.

Causes of Poor Coherence

Three main factors are the most common causes for a loss of coherence:

- A problem with the measurement system
- Environmental noise causing measurement signal contamination
- Reverberation

Problems with the measurement system

The most common measurement system issue affecting coherence is a timing mismatch between the reference and measurement signals. Loss of coherence from a timing mismatch will show up in the higher frequencies first, however the mismatch needs to be a significant fraction of the measurement time window to become obvious. Small timing issues may not be very visible when using large FFT sizes. The MTW transfer function uses very small time windows in the upper octaves and is therefore much more sensitive to small timing mismatches than large, single-FFT-size measurements. If you don't see coherence falling off more on the high end than at lower frequencies in an MTW measurement, then timing probably isn't the problem.

Other potential factors that could impact coherence on the measurement system end include excessive electronic noise, distortion, nonlinear processes such as compression and limiting, or crosstalk or other mixing of signals in the measurement system signal path, but these are less common.

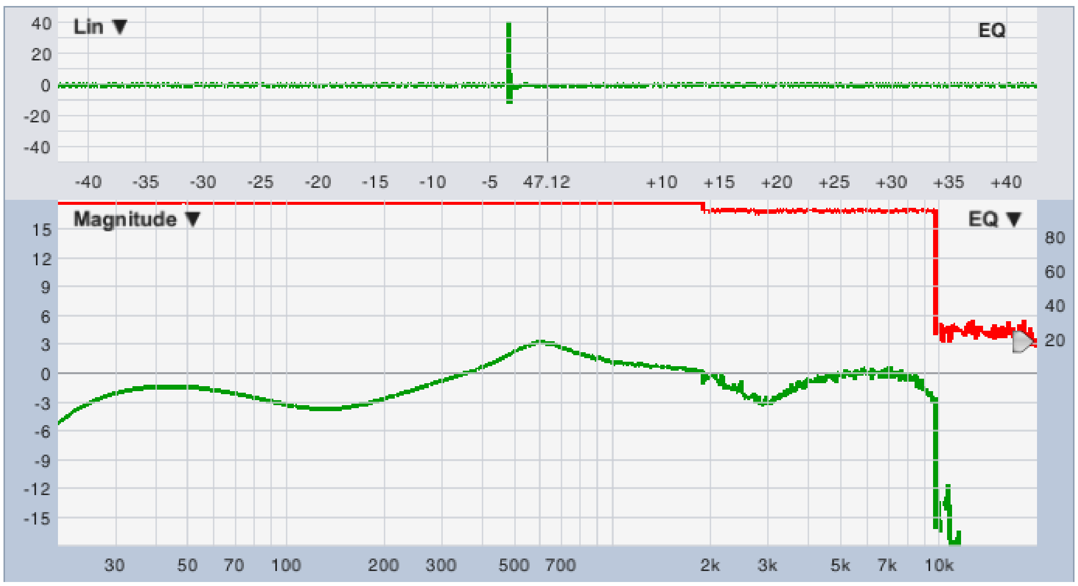


Figure 107: HF coherence loss in the MTW transfer function due to delay mismatch

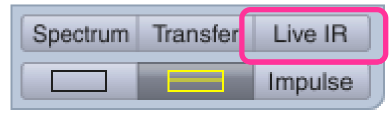
Environmental noise

Since coherence essentially works out to be an estimation of linearity/causality, any components of the measurement signal that are uncorrelated with the reference signal including background noise, HVAC noise, construction noise, people talking or shouting, etc., will have a negative impact on coherence. You know the problem is noise if measuring louder improves coherence. The solutions are to either measure louder, or possibly to reduce background noise if you can – for example by shutting off HVAC systems or asking people to take a break while you finish your measurement. Using more averaging will probably not improve the overall coherence level but it can have a stabilizing effect on the coherence trace.

Reverberation

If possible measurement system issues have been ruled out and coherence does not improve when you measure louder, the likely problem is reverberation. Reverberation is a non-linear phenomenon but increases proportionally when the excitation sound pressure level is increased – measuring louder will not improve the direct-to-reverberant sound ratio. There typically isn't much you can do about it unless you are in a position to damp reflective surfaces somehow, or perhaps re-aim a loudspeaker you are measuring so as to excite the reverberant field to a lesser degree.

Live IR



The transfer function *Live IR* graph displays the impulse response of the system under test – the time domain representation of its frequency response – continuously recalculated in real time. The live IR graph shows time on the *x* axis in milliseconds and amplitude or magnitude on the *y* axis, either as a percentage of digital full scale (Lin) or in decibels (Log or ETC), depending on the selected graph type shown in the upper left corner of the Live IR pane. The center point of the time axis is determined by the current measurement delay setting for a live transfer function measurement or the delay time stored in file for a captured data trace.

The *Live IR* pane is shown only when the *Live IR* button is engaged and one or both of the frequency domain transfer function graphs (*Magnitude* or *Phase*) is visible. Like the coherence display, the front trace on the *Magnitude* and *Phase* graphs determines what appears in the *Live IR* graph. The live IR is calculated only for live dual-channel measurements and IR data is included in captured traces only if the *Live IR* measurement was running at the time they were captured. That means no data appears on the *Live IR* graph if the front trace is a live average measurement, a captured snapshot of a live average, or a captured or imported transfer function trace that does not include IR data.

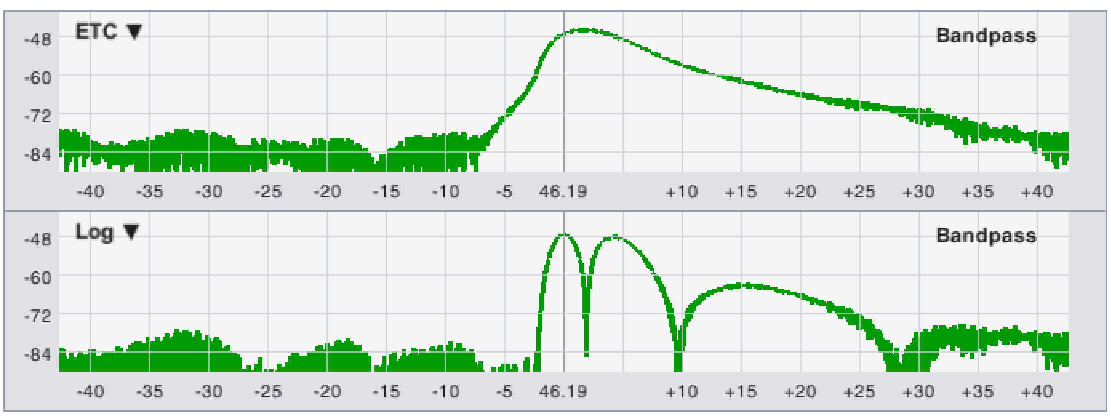


Figure 108: Log and ETC views of the impulse response of a low-frequency bandpass filter

The live IR is calculated independently of frequency domain transfer function measurements and so the *FFT* size and *Averaging* settings in *Measurement Config* do not affect it. The live IR FFT size (which determines its time constant) and the number of averages used are set in the *Live Impulse Response* section of *Transfer Function* options (*Options* menu > *Transfer Function*). The *Show LIR* check box in this section does the same thing as the *Live IR* button on the Control Bar in the main window. One additional setting on the *Transfer Function* options tab that affects the *Live IR* display is the *Proportional Panes* check box in the *Graph Settings* section. When this option is selected, the graph area in the main window is divided evenly between the *Live IR* graph and the other graph pane(s), rather than displaying it in a smaller, fixed size pane.

The three graph type options for the Live IR (selected via the menu in the upper left corner) are *Lin, Log* and *ETC*. *Lin* and *Log* display the impulse response with linear or logarithmic (dB) amplitude scaling. *ETC* displays the envelope time curve of the impulse response with decibel amplitude scaling. The *ETC* ends up looking like a smoother, less squiggly version of the *Log* IR that is often easier to read. The *Log* IR and *ETC* views are especially useful for looking at low frequency drivers and subs, where the peaks in the IR tend to be lower in level and spread out over a long time span, making them difficult to see on a linear amplitude scale particularly when measuring in a noisy environment.

Data Protection

We talked about coherence blanking on page 121 in the section covering the coherence display. Coherence blanking is one of several ways Smaart tries to keep bad or questionable measurement data off the screen and out of your decision making process. Some others include magnitude thresholding, overload protection, and signal present detection for the Live IR display.

Magnitude Thresholding

Magnitude thresholding works at the measurement level to ensure the validity of transfer function data. The idea is that if we did not put anything into the system under test at some given frequency then we should not be getting anything out, so Smaart looks at the level of the reference signal frequency by frequency and omits any frequency bins where the reference signal falls below the specified magnitude threshold when calculating the transfer function. Bins that fail the threshold test are simply not updated and so if a bin in question contains valid data from a previous measurement, Smaart leaves it alone. Frequencies that have never crossed threshold since the measurement began remain blank.

The *Magnitude Threshold* for transfer function measurements is user-definable. It is set from the *Graph Settings* section on *the Transfer Function* tab of the *Options* dialog (*Options* menu > *Transfer Function*).

Overload Protection

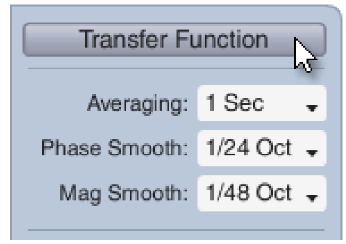
Overload protection applies only to transfer function and IR measurements. If you did the *Distortion* measurement exercise in the Spectrum measurements chapter then you saw that when we intentionally clipped the input signal, Smaart had no complaints about analyzing the spectrum of the clipped signal. Transfer function and IR measurements are a little pickier about their input data. If Smaart detects three or more consecutive samples with maximal amplitude values in either the reference or measurement signal, it assumes that clipping has occurred and will not use that data for transfer function or dual-channel impulse response measurement.

In IR mode if clipping is detected while recording data for a dual-channel measurement, Smaart will stop recording and throw an error message. In real-time mode, it throws away the buffer and gets a new one, and will keep doing this until it finds some unclipped data. If input levels are consistently overdriven, the measurement will freeze on the screen if it was already running when the problem occurred. If input levels are clipping when you start the measurement Smaart will not begin plotting data on the graph until the problem has been corrected.

Signal Presence Detection for IR Measurements

Signal presence detection for dual-channel measurements is similar to magnitude thresholding for frequency-domain transfer measurement. In this case however, Smaart simply stops processing the measurement when the reference signal falls below threshold. In IR mode if the reference signal is not present or is lost while recording data for a dual-channel measurement, Smaart will stop recording and throw an error message. In real-time mode, it will keep checking the input and begin or resume processing once the reference signal is acquired.

Transfer Function Options



We have probably covered most of the settings in *Transfer Function* options by now, but there are a few we didn't get to, so here is a complete listing of all the settings on the *Transfer Function* tab of the *Options* dialog with a brief description. To access *Transfer Function* options, you can click the *Transfer Function* label above the active measurement controls on the transfer function Control Bar in the main window, or you can select *Transfer Function* from the *Options* menu or press [Alt/Option]+[T] on your keyboard.

General Settings

The settings in this section are global measurement parameters for all frequency-domain transfer function measurements:

The *FFT* control sets the FFT size (in samples) for transfer function measurements. The FFT size, along with sampling rate, determines the time and frequency resolution of the measurement. The default setting is *MTW*, which generally works well for most applications. The other selections are conventional power-of-two FFT sizes.

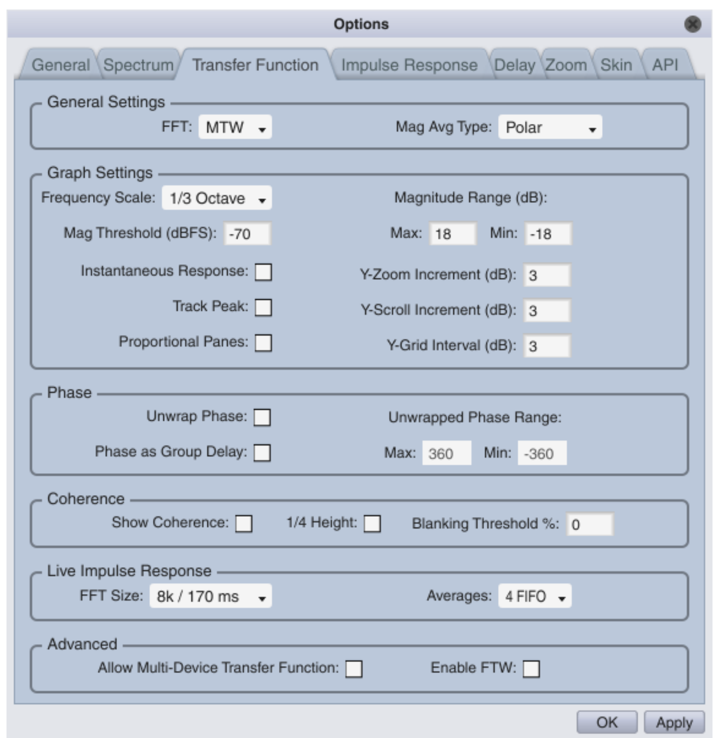


Figure 109: The Transfer Function options dialog page

Mag Avg Type determines the magnitude averaging type for all real-time frequency-domain transfer function measurements. There are two options for this setting:

- Complex averaging uses the complex transfer function data, maintaining two separate averages for real and "imaginary" data for each frequency, then converts the averaged results to decibel magnitudes for display after each new update. Complex averaging tends to reject reverberant energy as noise and may give you a better sense of speech intelligibility than polar averaging.

- *Polar* averaging converts complex transfer function data to decibel magnitudes before averaging, then averages the magnitude data. Polar averaging lets more reverberant energy into the average, which may tend to agree better with what you hear, particularly for musical program material.

Graph Settings

Frequency Scale determines the type of frequency scaling used for transfer function *Magnitude* and *Phase* displays. The options are linear (*Lin*) or logarithmic (*Log*). Actually, there are only two scaling options: linear (*Lin*) and logarithmic. The other choices are grid-ruling options for log-scaled frequency.

- *Decade* plots *Magnitude* and *Phase* graphs with logarithmic frequency scaling and decade (base 10) vertical grid ruling.
- *Octave* plots *Magnitude* and *Phase* graphs with logarithmic frequency scaling and vertical grid lines spaced at one-octave intervals.
- *1/3 Octave* plots *Magnitude* and *Phase* graphs with logarithmic frequency scaling and vertical grid lines spaced at 1/3-octave intervals.
- *Lin* plots *Magnitude* and *Phase* graphs with linear frequency scaling and vertical grid ruling.

Mag Threshold (dB FS) sets the minimum allowable reference signal level for transfer function measurements. At frequencies where the magnitude of the reference signal does not meet or exceed the value specified here the data is ignored and will not be added to the average. Instead, the value from the most recent update to cross threshold is held over until new data comes in to replace it.

Instantaneous Response displays instantaneous frequency response data for the front trace along with the (typically averaged) standard trace data. Instantaneous response, when enabled, will appear on the graph as unconnected dots rather than a line trace. Please note that this option can consume a lot of graphics processing resources and may result in slower performance on some machines.

Track Peak causes Smaart to track and display magnitude and frequency of the data point with the highest magnitude in the front trace the transfer function Magnitude plot when enabled.

Proportional Panes allows the *Live IR* graph to occupy an equal proportion of the graph area relative to other graph panes (rather than a smaller, fixed-height graph pane) when the *Live IR* graph is visible.

Magnitude Range (dB) sets the default decibel range for the transfer function *Magnitude* display.

Y-Zoom increment (dB) sets the increment used for keyboard zoom on the *y*-axis of the *Magnitude* graph. When a transfer function magnitude display is selected in the plot area pressing the [+/=] or [-] keys will increase or decrease the vertical scale of the graph by the number of decibels specified here.

Y-Scroll increment (dB) sets the increment for keyboard scrolling in the transfer function magnitude display. When a transfer function *Magnitude* display is selected in the plot area, each press of the up/down arrow keys will scroll the plot up or down by the number of decibels specified in this field.

Y-Grid Interval (dB) sets the grid ruling interval for transfer function *Magnitude* graphs.

Phase

Unwrap Phase "unwraps" the phase display when selected, by looking for "wrap" points where the phase trace crosses the +/- 180° boundary, then "splices" the trace at these wrap points to give you a more continuous view of phase response. Keep in mind however, that the actual phase data is always in the range +/- 180°, meaning the wrapped display has to rely on some assumptions that may be questionable in some cases. This type of display tends not to work very well if the incoming measurement data has a poor signal-to-noise ratio.

Phase as Group Delay, when selected, converts phase angles between adjacent frequencies in the phase display to relative time values (in milliseconds). A value of zero milliseconds for a given data point means the reference and measurement signals are arriving at exactly the same time at that frequency. Positive time values indicate that the measurement signal is arriving later than the reference signal at those frequencies. Negative time values indicate that the measurement signal is arriving before the reference signal. Be aware that time values for frequencies where the measurement is noisy may be questionable.

Unwrapped Phase Range sets the minimum and maximum values (in degrees) for the unwrapped phase display. You can also adjust the range of the both unwrapped phase and group delay graphs using the [+]/[−] keys on your keyboard or by rubber-band zooming with your mouse or other pointing device.

Coherence

Show Coherence displays coherence by frequency for the trace at the top of the *z*-axis on the transfer function Magnitude display. The coherence trace is shown in the upper half of the Magnitude display between 0 dB and the top of the graph. Note that because coherence is calculated by comparing averaged and un-averaged transfer function data, the coherence trace is not displayed when averaging is set to *None*.

1/4 Height: Checking this box compresses the coherence display, normally plotted in the upper half of the transfer function magnitude graph, to the top 1/4 of the graph. Select this option if you want more unobstructed graph area for magnitude traces when displaying coherence.

Blanking Threshold % sets the minimum allowable coherence value for transfer function Magnitude and Phase displays. Frequency data points with coherence values that fall below the value specified here will not be displayed.

Live Impulse Response

Show LIR displays the Live IR graph pane when a transfer function *Magnitude* or *Phase* display is visible in the main graph area. This control has the same effect as the *Live IR* button on the Control Bar in the main Smaart window.

FFT Size sets the FFT frame size (in samples) for the *Live IR* display. The resulting time constant, based on the current sample rate setting in Audio Options, is calculated and displayed for each available FFT size.

Averages sets the number of averages used for the *Live IR* display. Increasing this value may provide a more stable display at the expense of responsiveness. Increasing this value will also give you a better chance of capturing a usable Live IR measurement when measuring under difficult conditions.

Advanced

The controls in this section turn on advanced transfer function features that are turned off by default. We generally recommend that you leave these disabled when not in use.

Allow Multi-Device Transfer Function lets you select input channels from two different input devices as your reference and measurement signal sources for transfer function measurements. Please be aware that this will really only work if the sample clocks for the two devices are synchronized somehow, and even then you may encounter issues with relative delay time between the two devices changing when you stop and restart a measurement.

Enable FTW turns on Frequency-domain Time Windowing (FTW) for all transfer function measurements. FTW is a complex linear smoothing technique performed in the frequency domain that is mathematically equivalent to applying a tapered window function to the impulse response in the time domain and transforming the result with a zero-padded FFT. FTW is applied globally to all live and captured transfer function measurements that use Complex magnitude averaging (only). Note that checking the *Enable FTW* forces the global magnitude averaging selection (*Mag Avg Type*) for live transfer function measurements (see above) to *Complex*. When FTW is enabled, controls for turning it on and setting the nominal time window appear on the Control Bar in the main window. Please see *Live Measurement Controls,* beginning on page 73 for more information.

Application Example: Setting an Equalizer for a Loudspeaker

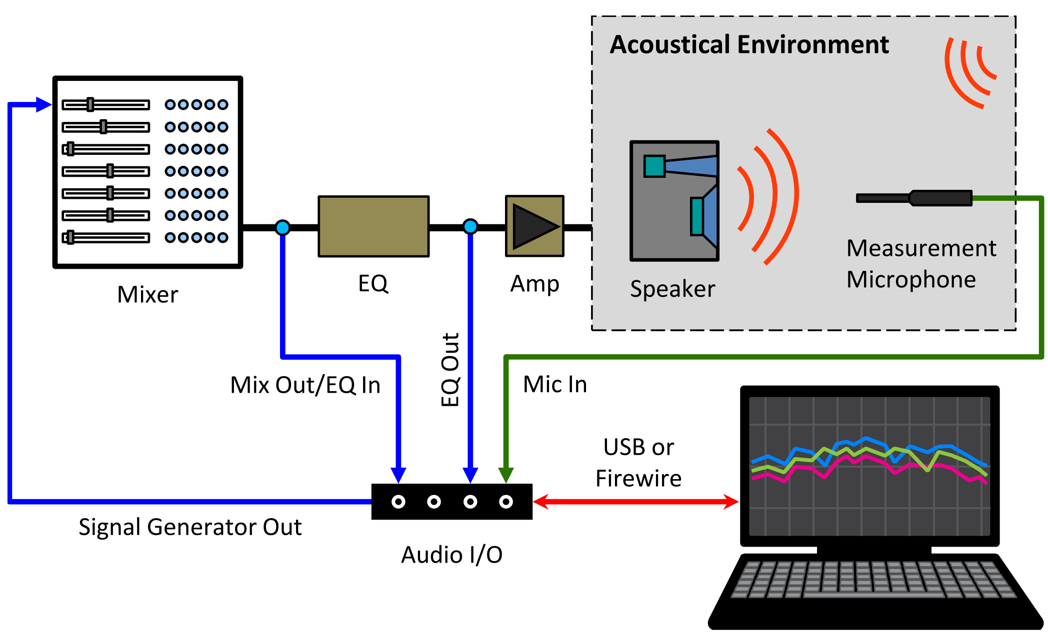


Figure 110: Measurement system setup for Setting an Equalizer for a Loudspeaker

In this example, we measure the Transfer Function of a loudspeaker and then adjust an equalizer to “flatten” its overall response. This example uses the hardware configuration shown in Figure 110. The setup shown enables us to simultaneously measure the equalizer (Mix Out compared to EQ Out in

Figure 110) and complete loudspeaker system (Mix Out compared to Microphone). If you do not happen to have a multi-channel I-O, this procedure can also be done sequentially, by measuring the loudspeaker and storing the measurement results, then re-patching to measure across the EQ while adjusting its filter settings, then measuring the loudspeaker again to check results.

To get started open Smaart and click the *Transfer* button at the bottom of the Control Bar that runs along the right side of the main Smaart window. This will split the graph area into two panes with a *Phase* graph loaded in the top pane and *Magnitude* below. It will also load the *Transfer Function* control set in the Control Bar, since both graph panes contain transfer function graphs. Click the button with the hammer and wrench icon next to the *Tab* selector in the Control Bar to open the *Measurement Config* page of the Configurator.

In *Measurement Config,* select the name of a tab in the tree view pane on the left, and then click the *New TF Measurement* button below the measurements table. Name your measurement “Mic One.” Select your audio I-O device on the *Device* selector and make the measurement signal channel (*Mea Ch*) the input channel that your microphone is on. The reference signal input (*Ref Ch*) should be set to the input channel connected to the output of your mixer.

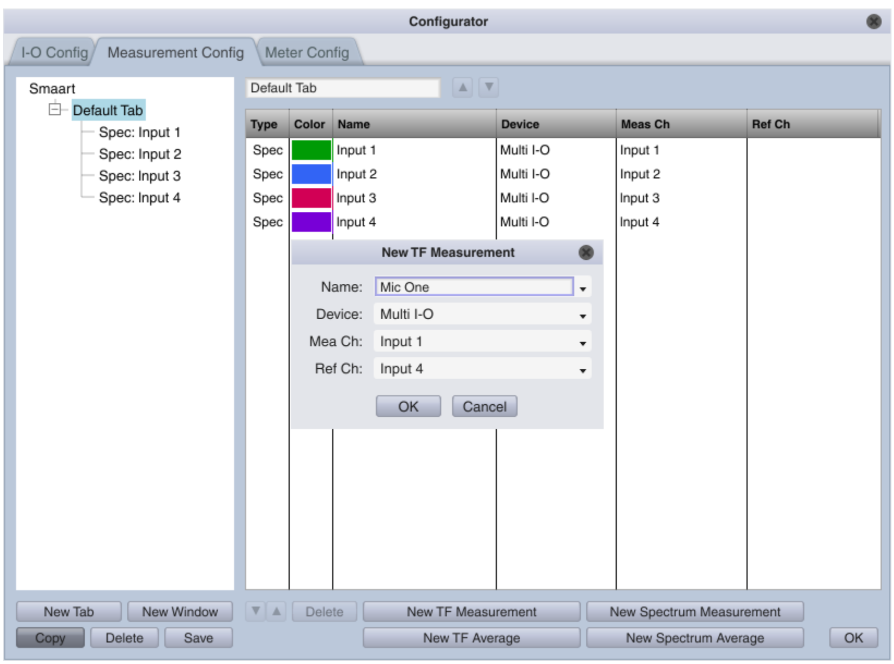


Figure 111: Creating a new transfer function measurement

If you don’t see your I-O device listed in the *Device* selector, cancel out of the dialog and click the *I-O Config* tab, and make sure its *Use* check box is checked in the devices table in the upper left and its *Status* is *OK*. Also, make sure that the *Use* check boxes for all the channels that you need are checked in the channels table below.

When you have made your selections, click *OK* to create the measurement and exit the *New TF Measurement* dialog. You should now see a *TF* measurement named *Mic One* at the top of the measurements table in *I-O Config*. Now, repeat this procedure to create a second new measurement named “EQ.” Its reference signal channel should be the mixer output in Figure 110 and its measurement channel should be the output of the equalizer.

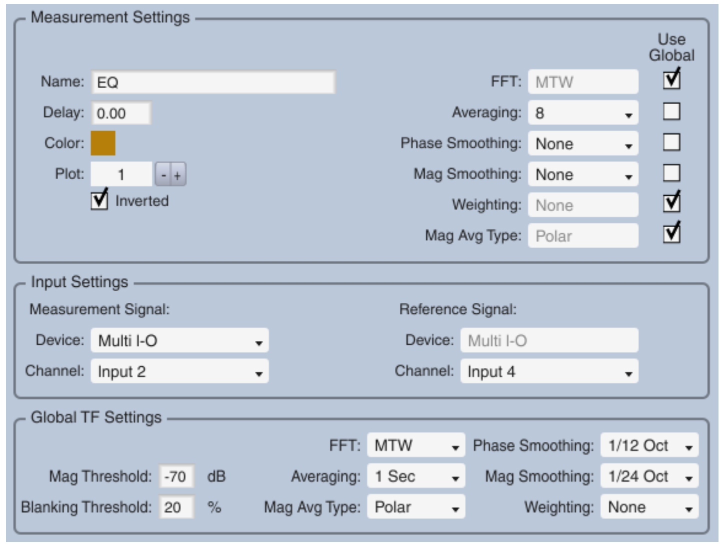


Figure 112: Measurement configuration for EQ measurement

Once that’s done, select the EQ measurement in the tree view pane or double-click its name in the measurements table to bring up its settings. In the Measurement settings section, un-check the *Use Global* check boxes for *Averaging, Phase Smoothing* and *Mag Smoothing*. Set *Averaging* to 8 FIFO or 16 *FIFO*, and both smoothing controls to *None*. Click the *Inverted* check box (to display inverse EQ response), then click *OK* to exit *Measurement Config*.

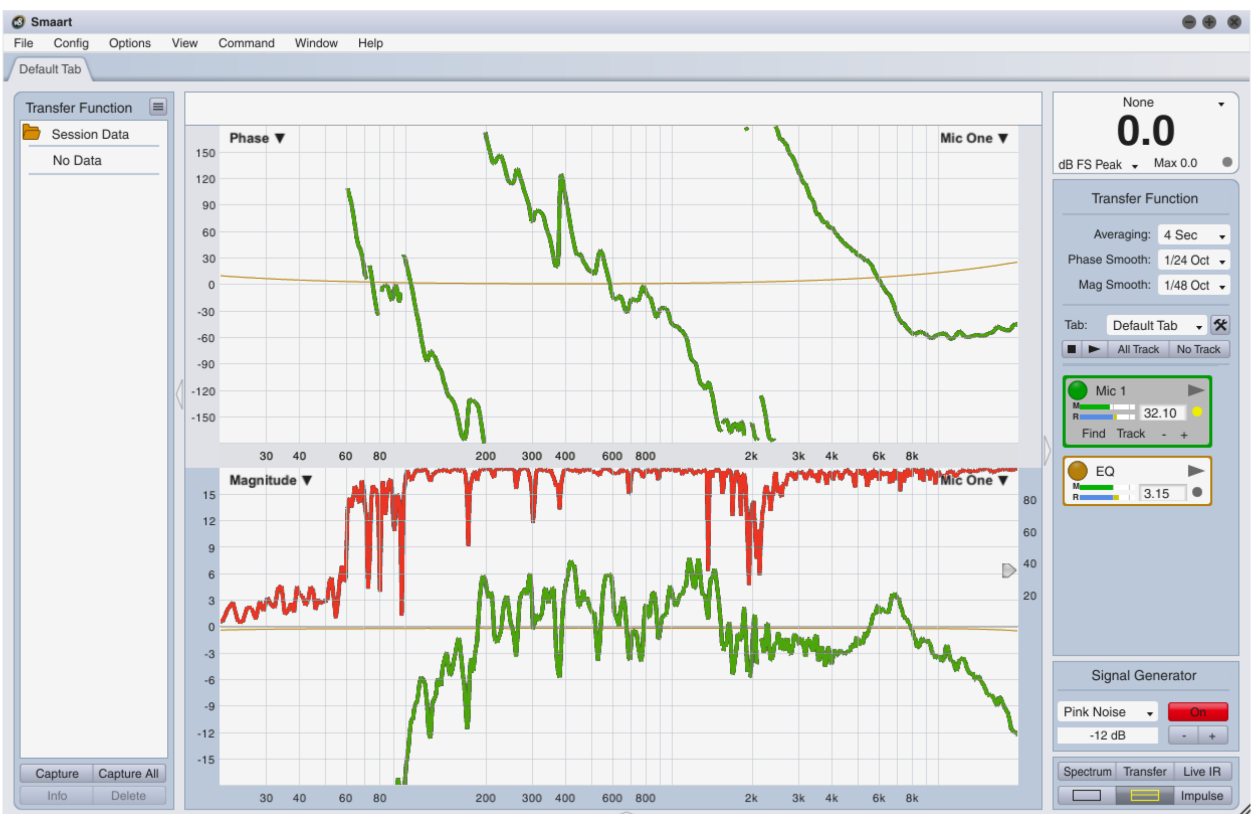


Figure 113: Loudspeaker measurement

Back in the main Smaart window, start the signal generator with *Pink Noise* selected and adjust the output level to a comfortable volume. Use the gain controls on your audio I-O device to adjust signal levels so that measurement and reference signal levels for the Mic One and EQ measurements are all running about equal, at a reasonable level and ensure that nothing is clipping. Click the start (▶) button for the Mic One measurement then click its *Track* button to find and set the measurement delay.

Set the *Averaging* control in the upper section of the Control Bar to something in the 2-4 sec range. Set *Phase Smooth* to 1/24 Oct, and *Mag Smooth* to 1/48 Oct. With any luck, your screen should look something like Figure 113 – your loudspeaker response curve will be somewhat different of course. Capture the loudspeaker response trace by pressing the spacebar on your keyboard. Name the captured trace "Pre EQ" (or any other name that makes sense to you).

Next, start the EQ measurement and if your equalizer is a digital device, use delay tracking to find and set the measurement delay time to compensate for its throughput delay. Once you have found the delay time you can turn tracking off. Since we set up the measurement to display the inverted EQ response you will note that cut filters make the EQ measurement trace go up and boost filters make it go down. Dial in a couple of respectably wide cut filters to match the major humps in your captured "Pre EQ" loudspeaker curve as we have in Figure 114. Notice the effect that they have on the live measurement of the loudspeaker response (Mic One).

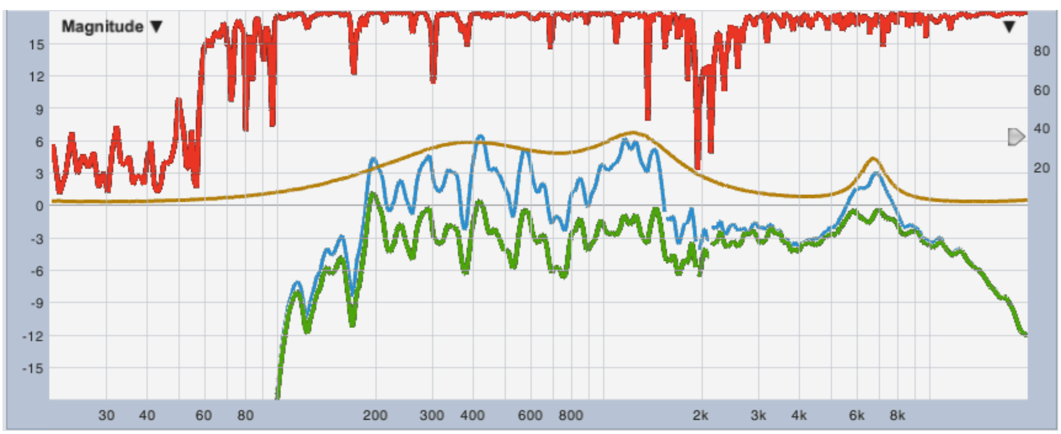


Figure 114: Transfer function measurement – initial loudspeaker response (Stored), EQ trace (inverted) and equalized loudspeaker response

That concludes this exercise. If you want to go for extra credit, try using music instead of pink noise as your reference signal, so that you can actually hear the effects of EQ settings changes as you analyze – we will leave it to the reader to figure out the signal source and routing for that.

Chapter 7: Impulse Response Measurement Basics

1: What is an Impulse Response?

In the most basic terms, an impulse response (IR) can be defined as the time domain (time vs amplitude) response of a system under test (SUT) to an impulsive stimulus. The word “system” in this case could mean something as small as a microphone or a single transducer, something as simple as a single filter on an equalizer. Or, it might mean something as big as a concert hall or sports arena, as complicated an entire sound system or a combination of the two. Smaart users of course are most often concerned with sound systems and their acoustical environments.

In the context of acoustical analysis, you might also think of an impulse response as the acoustical “signature” of a system. The IR contains a wealth of information about an acoustical system including arrival times and frequency content of direct sound and discrete reflections, reverberant decay characteristics, signal-to-noise ratio and clues to its ability to reproduce intelligible human speech, even its overall frequency response. The impulse response of a system and its frequency-domain transfer function turn out to be each other’s forward and inverse Fourier transforms.

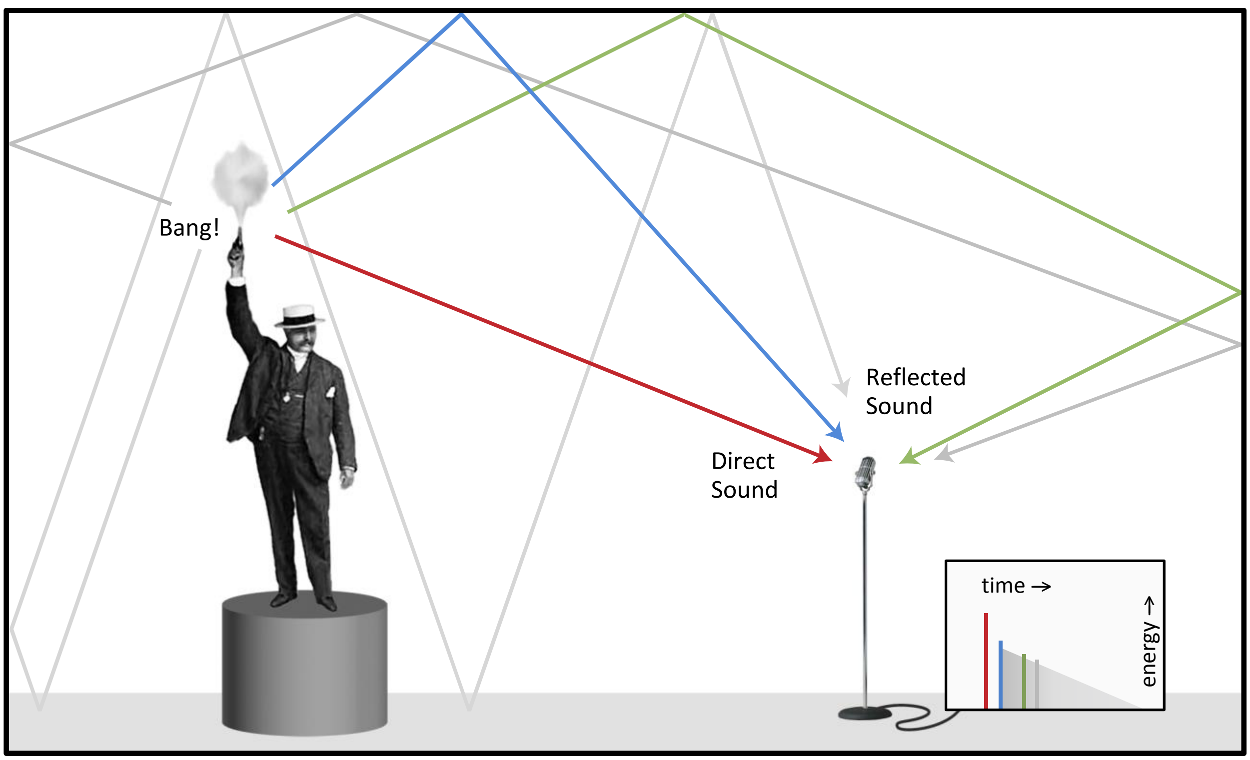


Figure 115: Conceptual illustration of an acoustical impulse. Sound from an excitation source arriving at a measurement position by multiple pathways, both direct and reflected. Here we see the path of direct sound from the source to the microphone in red, followed by a first order reflection in blue, a second order reflection in green, and higher order reflections in gray. Later arrivals tend to pile on top of each other forming a decay slope.

An acoustical impulse response is created by sound radiating outward from an excitation source and bouncing around the room. Sound traveling by the most direct path (a straight line from the source to a measurement position) arrives first and is expected to be the loudest. Reflected sound arrives later by a multitude of paths, losing energy to air and surface absorption along the way, so that later arrivals tend to come in at lower and lower levels. In theory this process goes on forever. In practice, the part we care about happens within a few seconds – perhaps less than a second in smaller rooms and/or spaces that have been acoustically treated to reduce their reverberation times.

The arrival of direct sound and probably some of the earliest arriving reflections will be clearly distinguishable on a time-domain graph of the impulse response. As reflected copies of the original sound keep arriving later and later, at lower and lower amplitude levels, they start to run together and form an exponential decay slope that typically looks like something close to a straight line when displayed on a graph with a logarithmic amplitude scale.

Anatomy of an Acoustical Impulse Response

Although no two non-identical rooms ever have identical impulse responses, there are a few component features that we can identify in some combination in almost any acoustical impulse response. These include the arrival of direct sound, early reflections, reverberant build-up and decay, and the noise floor. Figure 116 shows an acoustical impulse with its component parts labeled. Descriptions for each follow.

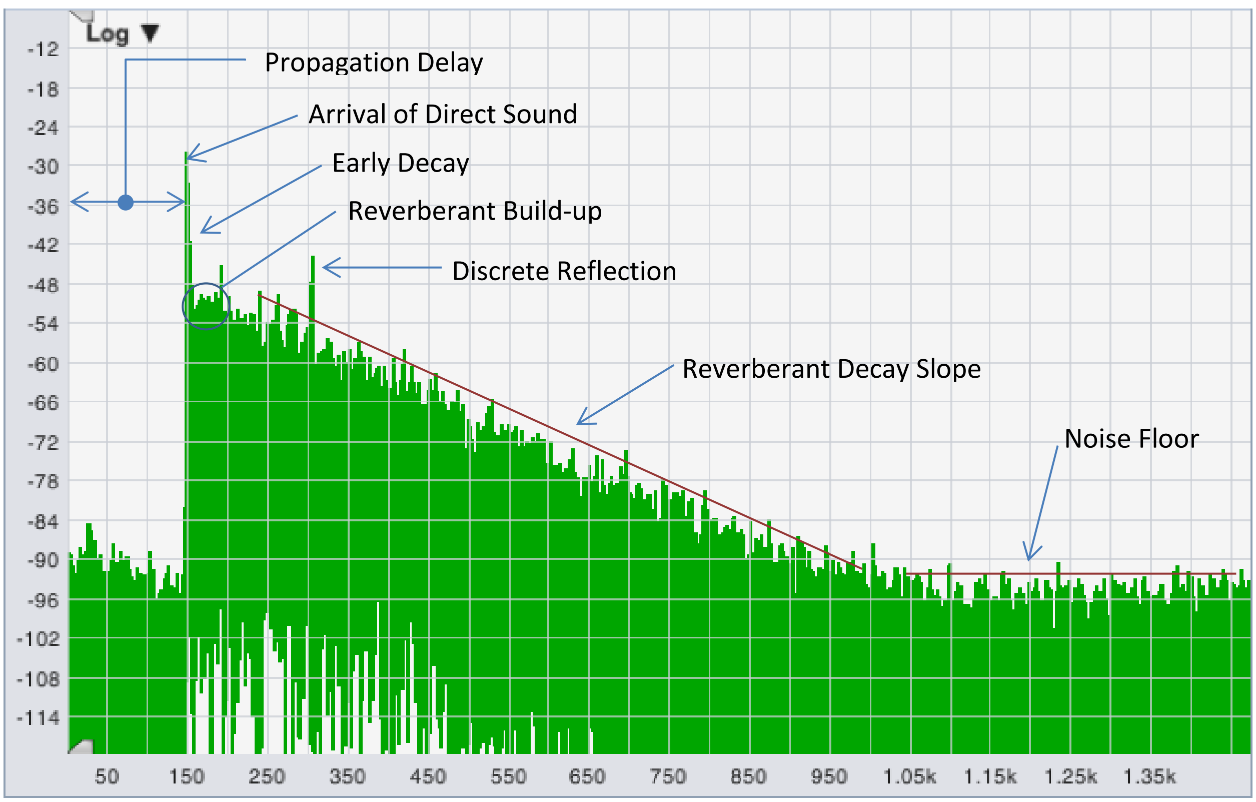


Figure 116: An acoustical impulse response with its common component parts labeled. This is a semi-log time domain chart with time in milliseconds on the x axis and magnitude in decibels on the y axis.

Propagation Delay

The time that it takes for direct sound from the sound source to reach the measurement position is the propagation delay time. This may include throughput delay for any DSP processors in the signal chain in addition to the time that it takes for sound to travel through the air.

Arrival of Direct Sound

Since the shortest distance between two points is always the straightest line, the first thing we expect to see when looking at an impulse response (IR) is the arrival of direct sound from whatever sound source we're using to stimulate the system under test. Depending on what we're trying to learn, the source could be an installed sound system, an omnidirectional loudspeaker brought in specifically for measurement purposes, a balloon pop or a shot from a blank pistol, or in a pinch, maybe hand claps or someone slamming a case lid shut.

In most cases, we would also expect the first arrival to be the loudest and correspond to the highest peak we can see in the IR, and in most cases we'd be right. There can be occasional circumstances where that might not turn out to be strictly true but in the vast majority of the cases, it should.

Discrete Reflections

After the arrival of direct sound, the next most prominent features we tend to see are sound arriving by the next most direct paths; the lowest order reflections. Sound that bounces off one surface to get from the excitation source to a measurement position is called a first-order reflection, two bounces gives you a second order reflection and so on. Reflected sound can be useful or detrimental, depending on factors such as its relative magnitude and timing in relation to the direct sound and the extent to which it is clearly distinguishable from the diffuse reverberant sound.

Early Decay, Reverberant Build-up, and Reverberant Decay

Following the arrival of direct sound and the lowest order reflections, sound in a reverberant space will continue bouncing around a room for a while, creating higher and higher order reflections. At any given listening position, some of this reflected energy will combine constructively over a relatively short period of time, resulting in a build-up of reverberant sound, before air loss and absorption by the materials that make up reflecting surfaces begins to take their toll. At that point, the reverberant decay phase begins.

In practice, you may or may not be able to see the reverberant build-up in an impulse response as distinct from the direct sound and early reflections. Sometimes it can be quite clearly visible, other times not so much. By convention, the first 10 dB of decay after the arrival of direct sound in the reverse-time integrated IR (we will get to that in *Chapter 9: Analyzing Impulse Response Data*) is considered to be early decay. Reverberant decay is conventionally measured over a range from 5 dB below the level of direct sound down to a point 30 dB below that on the reverse integrated IR, or 20 dB down in a pinch.

Noise Floor

In theory, the reverberant decay phase of the IR continues forever, as an ideally exponential curve that never quite reaches zero. In practice it reaches a point relatively quickly where we can no longer distinguish it from the noise floor of the measurement. Noise in an IR measurement can come from several sources, including ambient acoustical noise, electrical noise in the SUT and the measurement system, quantization noise from digitizing the signal(s) for analysis, and artifacts from DSP processes used for analysis.

Uses for impulse response measurement data

Delay Time Measurement

The most common use for impulse response measurements in Smaart is in finding delay times for signal alignment in transfer function measurements and for aligning loudspeaker systems. Each time you click the delay locator in Smaart an IR measurement runs in the background. In this case all we really care about is the initial arrival of direct sound, which is typically so prominent that you can pick it out with high confidence even when signal-to-noise ratio of the IR is poor, so we don't even bother displaying the results. Smaart simply scans for the highest peak and assumes that to be the first arrival, and most of the time that works very well.

Occasions where automatic delay measurements might not work well include measurements of low-frequency devices or any case where you're trying to measure a directional full-range system well off axis, in a location where a prominent reflection can dominate the high frequencies. In the latter case, it's possible for reflected HF energy to form a higher peak later than the arrival of direct sound, requiring you to visually inspect the IR data to find the first arrival.

Reflection Analysis

Another common use for IR measurements is in evaluating the impact of problematic discrete reflections. Reflected sounds can be beneficial or detrimental to a listener's perception of sound quality and/or speech intelligibility, depending on a number of factors. These factors include the type of program material being presented (generally, speech or music), the arrival time and overall level of the reflected sound relative to the level of direct sound, and the frequency content and the direction from which they arrive. As a general rule, the later they arrive and the louder they are (relative to direct sound) the more problematic they tend to be.

Reverberation Time (T60, RT60…)

Reverberation time is kind of the grandfather of quantitative acoustical parameters. First proposed by Walter Sabine a century ago, T60 or RT60 reverberation time is the time that it takes for reverberant sound in a room to decay by 60 decibels from an excited state (after the excitation signals stops). It is one of the most widely used (and in some cases perhaps misused) quantities in room acoustics. Although it is quite possible for two rooms with identical reverberation times to sound very different, when evaluated band-by-band it can still give you some idea as to the overall character of the reverberant field in a given room. In concert halls it can give you an idea of perceived warmth and spaciousness for music. In auditoriums, it is often used as a rough predictor of speech intelligibility.

Early Decay Time (EDT)

Early decay time ends up being the decay time for direct sound and earliest, lowest-order reflections. Since the earliest reflections tend to be the most beneficial in terms of separating sounds we want to hear from reverberation and background noise, EDT can give you some clues about overall clarity and intelligibility in a room and/or system. EDT, like RT60, is conventionally normalized to the time it would take for the system to decay 60 dB at the measured rate of decay.

Early-to-late energy ratios

Early to late energy ratios are a direct measure of the sound energy arriving within some specified interval following the arrival of direct sound, vs the energy in the remaining part of the IR. These provide a more direct method of evaluating the relationship between beneficial direct sound and early reflections that a listener hears versus the amount of (potentially detrimental) reverberation and noise, than inferences made from the early and reverberant decay rates.

Speech Intelligibility Modeling

Early to late energy ratios such as C35 and C50 have long been used as objectively measurable predictors of subjective speech intelligibility. In the 1970s Victor Peutz came up with Articulation Loss of Consonants (ALCons), a predictive metric for intelligibility based on the volume of a room and its reverberation time, the directivity of loudspeakers and distance from source to the listener. Later on, Peutz revised the equation to use a direct-to-reverberant energy ratio in place of volume, distance and loudspeaker Q, making ALCons a directly measureable quantity. More recently, the speech transmission indexes (STI and STIPA) have emerged as metrics that are generally more robust. All of these can be calculated from the impulse response of a system.

Chapter 8: Impulse Response Mode User Interface

If you already know your way around IR mode in Smaart 8 you can probably skip this section, but it might not hurt to at least skim over it. If you are new to IR mode then introductions are in order. To get to IR mode in Smaart, select *IR Mode* from the *View* menu, press the "I" key on your keyboard or click the Impulse button that appears in the lower right corner of the main window in real-time mode (just below the signal level /sound level meter). You will find yourself confronted with a screen like the one below. (The colors may be darker but the layout is the same.) On the right side of the window is a vertical strip of controls. The rest of the window is devoted to the graph areas and cursor readout.

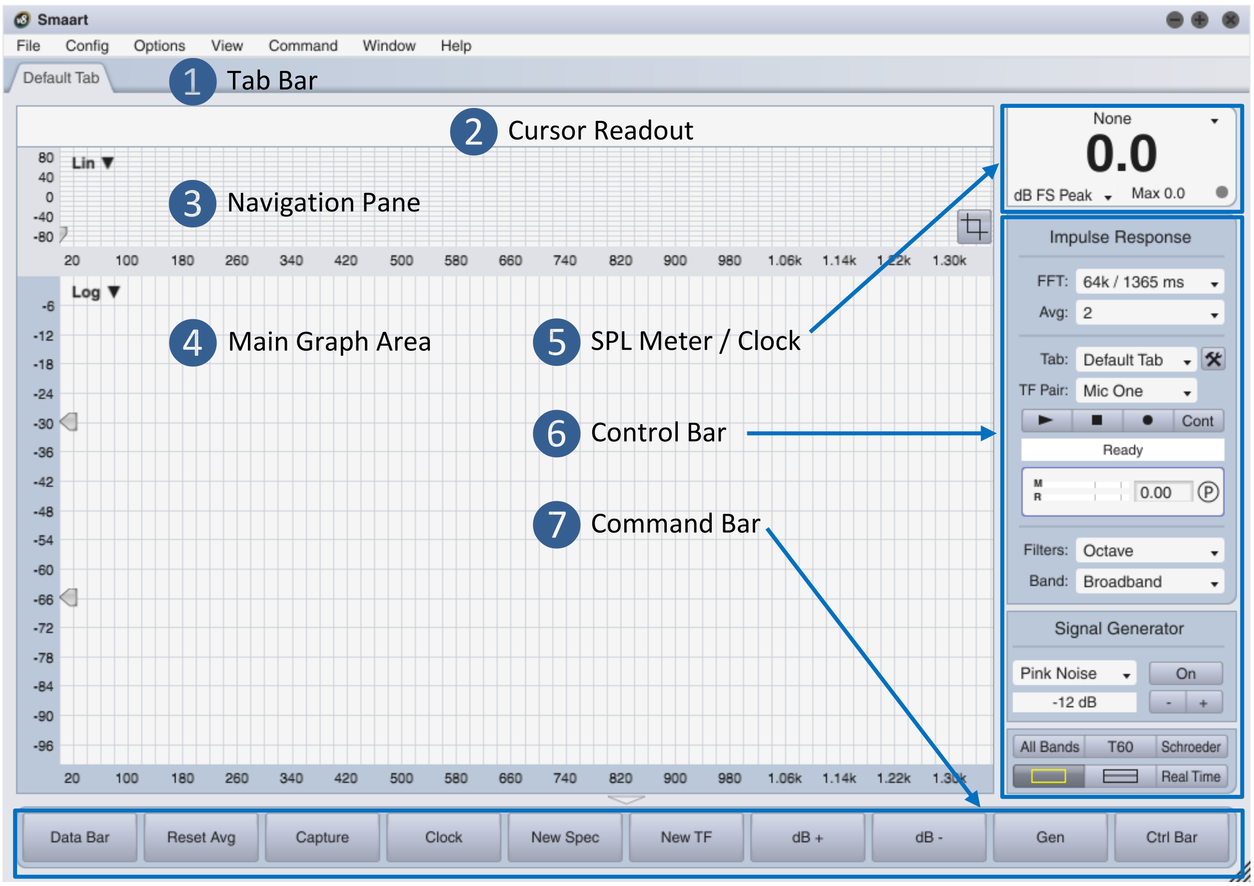


Figure 117: Anatomy of the main Smaart window layout in Impulse Response mode

❶ Tab Bar

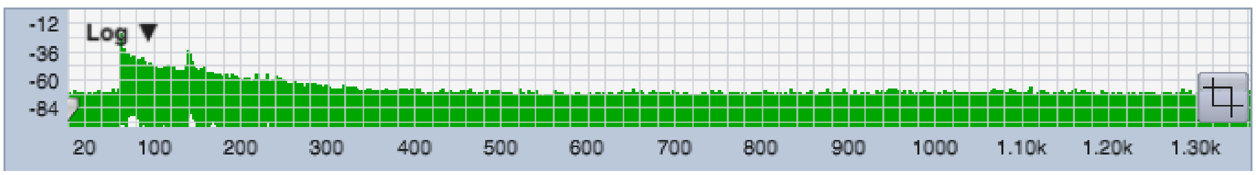
Smaart can run in multiple windows and each window can host multiple tabbed workspaces that we refer to simply as tabs. Each tab includes its own measurements, screen layout, and plot assignments and you can switch between them by clicking the tab-shaped buttons below the menu bar in the area we refer to as the Tab Bar. You can move a tab from one Smaart window to another by clicking on its button in the Tab Bar with your mouse and dragging it to another window, then releasing the mouse button to drop it.

❷ Cursor Readout

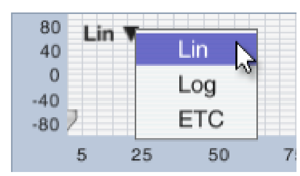
When measurement data is present, the cursor readout displays numeric coordinates for the cursor location(s) as you move your mouse over the graphs areas. Numeric coordinates are provided here for the cursor location in units of time, amplitude/magnitude and frequency, as applicable to graph type.

For time domain graphs (Lin, Log or ETC) in the main graph area(s), there are three sets of coordinates as show above. From left to right, they are the location of the locked cursor which typically marks the highest peak in the impulse response, the movable (mouse) cursor coordinates, and in brackets on the right, the difference between the locked and movable cursors. Note that time coordinates can optionally be displayed as both time (in milliseconds) and equivalent distance traveled, based on the currently specified speed of sound.

❸ Navigation Pane

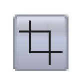


The small time-domain display in the upper part of the graph area is used for navigation and is always visible. Right-clicking and dragging (Ctrl + click and drag on Mac) across the graph in this pane selects a specific time range for display on the larger time-domain charts. The full IR time record remains visible in the navigation pane when you are zoomed in (unless you use the crop function). Clicking anywhere in the left margin of the plot clears the zoom range and returns any time-domain graphs in the main display pane(s) to the full IR time record.

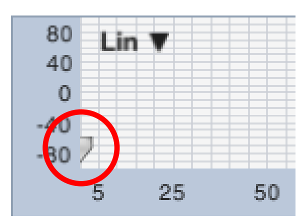


The selector control in the upper right corner of the navigation pane selects the graph type to be displayed in this area. The navigation pane is limited to time-domain graph types only (Lin, Log or ETC).

The *Crop* button in the lower right corner of the navigation pane can crop a file for display purposes to show only the selected time range – a very useful feature when working with IR measurements with long noise tails. Cropping is non-destructive and can be undone – clicking the Crop button again on a cropped measurement restores the full extent of the original time record – however if you save the IR to a file while cropped, the cropped version is written to file.



The little arrow shaped widget positioned in the lower left corner of the navigation pane (circled in red in the screen clip show the right) is the Time 0 marker. When you record a dual-channel impulse response measurement in Smaart, this marker is automatically set to match the reference signal delay time – if you are familiar with real-time transfer function measurement it's analogous to the center point of the Live IR. For single-channel measurements



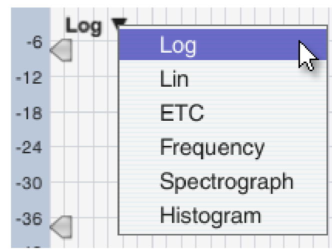
or file-based data it is set to the beginning of the time record. Dragging it to the left or right moves the time-zero point for all time-domain graphs (Lin, Log or ETC) displayed in the main graph area.

❹ Main Graph Area

The larger (lower) portion of the graph area can be divided into one or two panes by clicking the buttons labeled with rectangles in area #5. Each pane can host any one of the six available main graph types:

- *Lin*, *Log*, *ETC* (time domain views)
- *Spectrograph* (frequency and level vs time),
- *Frequency* (spectrum)
- *Histogram* (bar chart of quantitative acoustics values by octave or 1/3-octave band.)

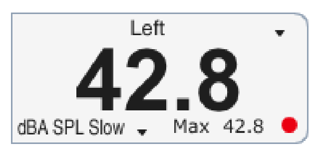
We will discuss each of these display types in the next chapter.



The graph type for each pane is selected by means of the drop-list control in the upper left corner of the pane. Some main graph types also have additional selector controls in their upper right corner that control display options specific to that graph. The two arrowhead-shaped widgets to the left of the list of graph types appear on all time-domain and spectrograph plots. They control the dynamic range for the spectrograph.

As you probably noticed, the main graph area and the navigation pane are blank when you first enter IR mode. You won't see any data in the graph areas until you record a measurement or load an impulse response from file (*File* menu > *Load Impulse Response*). Smaart can open and analyze .wav and .aiff files containing any type of audio data, but IR mode is purpose-built for analyzing impulse responses. There is no multi-channel file support or optimization for working with files more than a few seconds in length – and of course a lot of the IR analysis capabilities are irrelevant for other types of audio data.

❺ SPL Meter / Clock



The large numeric readout that appears (by default) at the top of the Control Bar in the upper right corner of each tab can be configured to function as a Sound Pressure Level (SPL) meter, an integrating Equivalent Sound Level (Leq) meter, a peak signal level meter calibrated to normalized digital full scale, or a clock. When the level meter is displayed, pressing the [K] key on your keyboard switches the display to a clock and vice versa. This display can be hidden if you don't need it by selecting *SPL Meter* from the *View* menu pressing [Alt/Option] + [K] on your keyboard. When hidden, repeating either of these actions will restore it.

The in-tab SPL Meter operates almost identically to the meter module in the *SPL Meters* window. Both are covered in detail in the section on *Sound Level Metering* on page 40. Note that in order to perform accurate SPL or Leq measurements, the input being monitored must be calibrated to SPL. Please see *Sound Level Calibration* on page 67 for more information.

❻ Control Bar

The control bar in impulse response mode includes live measurement controls, bandpass filters signal generator and main display controls

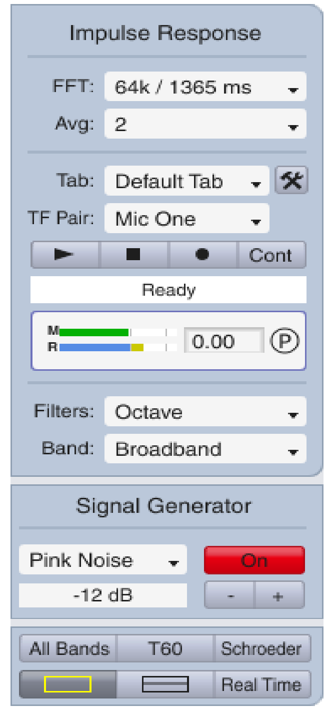


Figure 118: Impulse response mode Control Bar

Live Measurement Controls

The upper section of the Control bar is dedicated to live impulse measurement controls for recording impulse response measurements in the field. Note that you can also analyze file-based data in impulse response mode. To load and impulse response from a .wav or .aiff file, select *Load Impulse Response* from the *File* menu and then browse to the location of the file that you want to open in the *Load Impulse* dialog and open it.

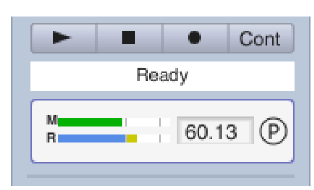
The *FFT* size and averaging (*Avg*) controls together determine the measurement duration for dual-channel IR measurements. Notice that for each FFT size, the time constant is given along with the FFT size in samples. The FFT time constant, also called the time window, is the amount time it takes to record the required number of samples at the currently selected sampling rate.

Averages (*Avg*), sets the number of successive IR measurements to average together to improve the signal-to-noise ratio of dual-channel measurements. For deterministic IR measurements made using period-matched signals, the number of averages is normally set to a low number or even “None”. When measuring with random signals or in noisy environments, more averaging can greatly improve the signal-to-noise ratio of impulse response measurements.

The *Tab* selector can be used to switch between tabs if the *Tab Bar* is hidden. The button to its right labeled with the hammer and wrench icon opens the *Measurement Config* page in the *Configurator* dialog.

TF Pair selects the signals to be used for dual-channel IR measurements. Dual-channel IR measurements in Smaart are essentially transfer function measurements with the addition of an inverse Fourier transform at the end, and any of the reference and measurement signal pairs that you have set up in the current tab for real-time transfer function measurements may be used for IR measurements. Just click on this control and select the name of a measurement that uses the input channel that you want to record. For single-channel recordings, only the measurement signal channel is recorded.

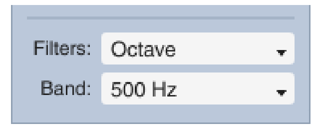
The live measurement controls in IR mode are analogous to a transfer function “measurement engine” in real-time mode, with a couple of extra twists. Starting from the top left, the buttons marked with a triangle (▶) and a square (■) start and stop a measurement.



The button labeled with a circle (●) works like the record button on a tape deck or digital recorder, but in this case, it is a measurement *mode* control. Clicking the start (▶) button *without* the record button punched in kicks off a dual-channel IR measurement. With the record button (●) activated, Smaart becomes a single-channel digital recorder and records just the measurement signal channel of the selected signal *TF Pair*. The idea is that you would start the recording and pop a balloon or fire your starter pistol (or whatever), and then click the stop (■) button to end the recording and display your results.

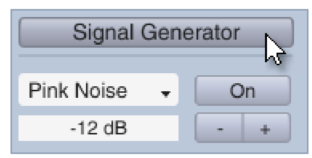
The Continuous (*Cont*) button causes the dual-channel measurement routine to run continuously, starting over again automatically each time it finishes a measurement until you tell it to stop (by clicking the stop button). The results of the last measurement are displayed while it's recording and processing the next measurement. Click the stop button (■) to end the recording and display the recorded data.

Bandpass Filters



The broadband impulse response is useful for finding delay times and discrete reflections, but for most acoustical analysis purposes, the IR needs to be filtered into octave bands or sometimes 1/3-octave bands. Smaart includes complete sets of octave and 1/3-octave bandpass filters for impulse response analysis. Bandpass filtering is non-destructive and is done on the fly whenever you need it. All that you have to do is select the filter set that you want to use (Octave or 1/3-Octave) using the *Filters* selector on the Control Bar and then choose the center frequency for the band that you want to analyze from the *Band* list.

Signal Generator Controls



The next group of controls on the Control Bar is for the signal generator. The label at the top of this section is actually a hover button (it turns into a button when your mouse cursor passes over it) that opens the *Signal Generator* dialog, which contains a lot more options for the signal generator than we could fit on the Control Bar. Below are a signal type selector (*Pink Noise* is selected in the example to the right) and an output level field that shows the current output level in normalized dB full scale. The *On* button turns the generator on or off – it glows an angry red when the generator is running. The minus and plus (-|+) buttons to the right of the output level field bump the signal level down or up by 1 dB.

Main Display Controls

The last group of controls in the control strip on the right side of the real-time mode window is devoted to data display functions. Starting from the top left of the screen clip shown here on the right:

- The *All Bands* button opens the All Bands table, where you will find nearly all of the quantitative acoustical metrics that Smaart can calculate automatically for an impulse response. See *Histogram and All Bands Table* for more in this feature.
- Clicking the *T60* button displays level marker widgets used for calculating reverberation time and early decay time on Log IR or ETC plots.
- The *Schroeder* button displays a reverse time integration curve on Log IR or ETC plots.
- The two buttons labeled with rectangles divide the main plot area into one or two graph panes: One rectangle, one pane; two rectangles, two panes.

- The *Real Time* button exits IR mode and takes you back to real-time frequency domain measurement mode. (In real-time mode it changes to an *Impulse* button that will bring you back to IR mode.)

7 Command Bar

The Command Bar is a user-configurable button bar that runs across the bottom of a Smaart window. You can hide and restore it by clicking the triangular button in the border area just above it. This show/hide button remains visible in the window border when the Command Bar is hidden and clicking it again will restore it to visibility. You can also hide/restore the Command Bar by selecting *Command Bar* in the *View* menu or by pressing the [U] key on your keyboard. To customize the command bar, select *Command Bar Config* from the *Config* menu (see *Configuring the Command Bar* on page 38 for details).

Additional Options for Impulse Response Measurement and Analysis and Display

Some additional options pertaining to IR measurement that don't appear on the main screen can be found in the *General* and *Impulse Response* options pages, accessible by selecting *General* from the *Options* menu or pressing [Alt/Option] + [O] on your keyboard.

General Options Pertaining to Impulse Response Displays

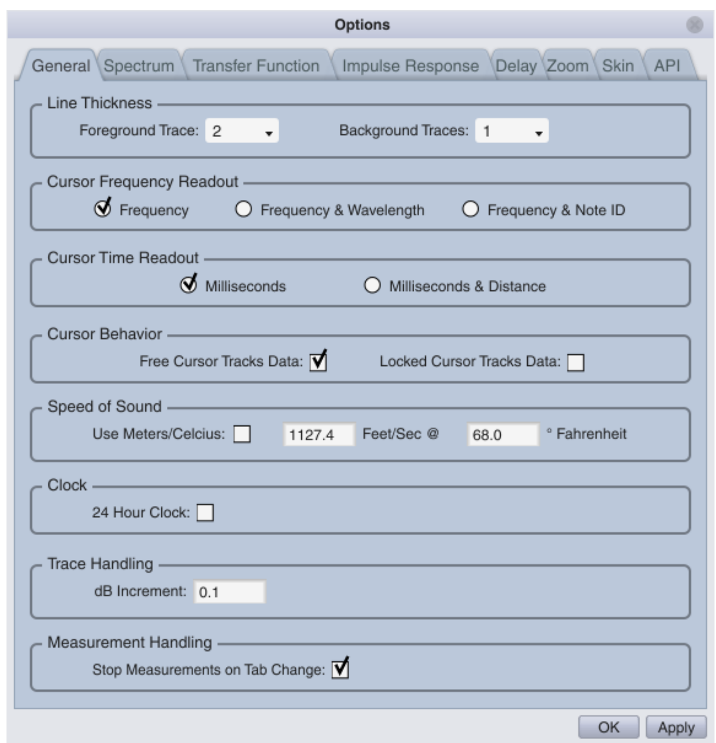
Cursor Time Readout

The cursor time readout setting applies to the both IR Mode and the Live IR graph on Transfer Function displays in Real-Time mode.

- *Milliseconds* – displays time coordinates and relative time differences in milliseconds only.
- *Milliseconds & Distance* – displays time coordinates as milliseconds and equivalent distance, based on the Speed of Sound settings.

Speed of Sound

The settings in this section determine the speed of sound that Smaart uses for calculating equivalent distances for time coordinates and also whether distances are displayed in feet or meters. It can also serve as a handy speed of sound calculator any time you need to know the speed of sound for a given air temperature.



- *Use Meters/Celsius* – when this option is selected, Smaart displays distances in meters and the temperature used for calculating speed of sound in degrees Celsius. Otherwise Smaart displays distances in feet and uses degrees Fahrenheit for temperature.
- *Speed of sound* ([unit]/sec) and temperature – At elevations where humans are comfortable breathing, the speed of sound is mainly a function of temperature, and so the two inputs are linked. Changing the temperature setting automatically recalculates the corresponding speed of sound and vice versa.

Impulse Response Options

Most of the controls in the Impulse Response options dialog tab echo the settings of the controls in the main window. We have already discussed those in some detail so we will concentrate here on the ones that do not.

Time Domain Display Settings

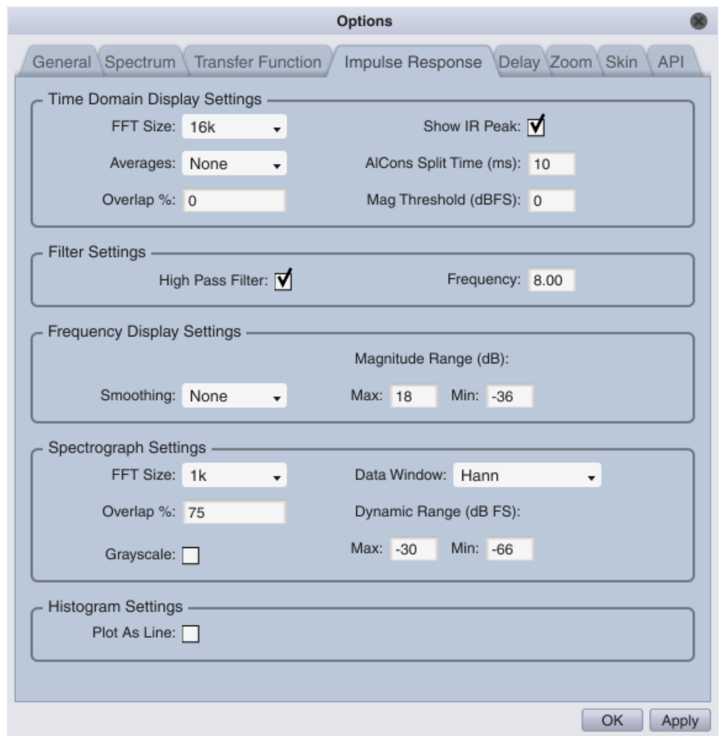


Figure 119: Impulse Response options

- *FFT Size* and *Averages* echo the settings of the on-screen controls for dual-channel IR measurements that we talked about on page 140.
- *Overlap %* (for Averaging, not to be confused with overlap for the *Spectrograph* display) – When overlap is set to any value other than zero, each successive measurement going into an averaged dual-channel IR measurement shares the specified percentage of data with the previous frame(s).
- *Show IR Peak* sets the locked cursor in IR mode to the highest peak in the impulse response each time you run a new measurement.
- *ALCons Split Time* sets the split time for the early-to-late energy ratio used in calculating ALCons (a type of speech intelligibility estimation that can be calculated from an impulse response). There is no real standard for this parameter but common settings are 10 or 20 milliseconds.
- *Mag Threshold (dBFS)* is similar to magnitude thresholding in transfer function measurements. It is turned off when set to zero. When set to any other value (in dB FS), Smaart will zero out the transfer function at any frequency where the reference signal does not cross threshold before calculating the (dual-channel) impulse response.

Filter Settings

Smaart includes a sweepable highpass filter for IR measurements that can be handy when analyzing IR measurements that include a lot of very low-frequency noise or in cases where the reference signal being used in a dual-channel measurement is band-limited somehow. The filter is applied post process to IR data for display purposes – meaning it can be used for file-based or newly measured data – and only affects what you see on the screen. It does not change the underlying measurement. The cutoff frequency for the filter can be set to any value between 0 Hz and one half of the Nyquist frequency (equal to one half of the sampling frequency) for the currently selected audio sample rate.

Frequency Display Settings

- *Smoothing* echoes the setting of the Smoothing control in the upper right corner of the *Frequency* graph in IR mode.
- The *Magnitude Range (dB)* controls are useful for setting a specific decibel range for the Frequency graph in IR mode, however you can also resize the range using the +/- keys or by right-click-and-drag mouse zooming, as you can with other graph types in Smaart.

Spectrograph Settings

- The *FFT Size* and *Overlap* controls echo the settings of the controls found in the upper right corner of the Spectrograph display in IR mode. Together they determine the time resolution of the spectrograph.
- *Grayscale* plots the spectrograph using varying shades of gray instead of color to represent magnitude.
- *Data Window* – sets the data window function used in calculating the individual FFTs used to create the spectrograph display. You can leave this set to Hann unless you have some good reason to change it.
- *Dynamic Range* echoes the settings of the slider control widgets found on the left edge of time domain and spectrograph displays on IR mode. The spectrograph scales its color (or grayscale) spectrum to the range between the Min and Max values and plots decibel values above the Max thresholds in white and below the Min in black.

Histogram Settings

The *Histogram* chart in IR mode plots the values found for any column in the *All Bands* table band-by-band for all octave or 1/3-octave bands. Selector controls are found in the upper right corner of the chart in the main window. By default, Smaart plots the histogram chart as a bar graph. Selecting *Plot as Line* in *Histogram Settings* causes this chart to be plotted as a line graph instead.

Chapter 9: Analyzing Impulse Response Data

Smaart provides a powerful set of tools for analyzing impulse response data in both the time and frequency domains. Time-domain analysis tools include logarithmic and linear time-domain views, Energy Time Curves, octave and 1/3-octave bandpass filters, reverse time integration and automatic calculation of common acoustical parameters such as EDT, RT60 and clarity factors. Frequency domain analysis tools include spectrum analysis of arbitrary time ranges and the Spectrograph.

If we were discussing Smaart's real-time measurement and analysis mode, we would almost have to pause at this point to set-up and start actively measuring some kind of sound source in order to have something to analyze. But in IR mode, measuring and analyzing are generally two separate things that we can talk about separately. Data analysis in IR mode is an off-line, post-process affair that works the same whether we're onsite actively measuring a system or working with an impulse response recorded in a .wav or .aiff file. Since we just talked about the IR mode user interface in the previous chapter, let's dive right into actually using it.

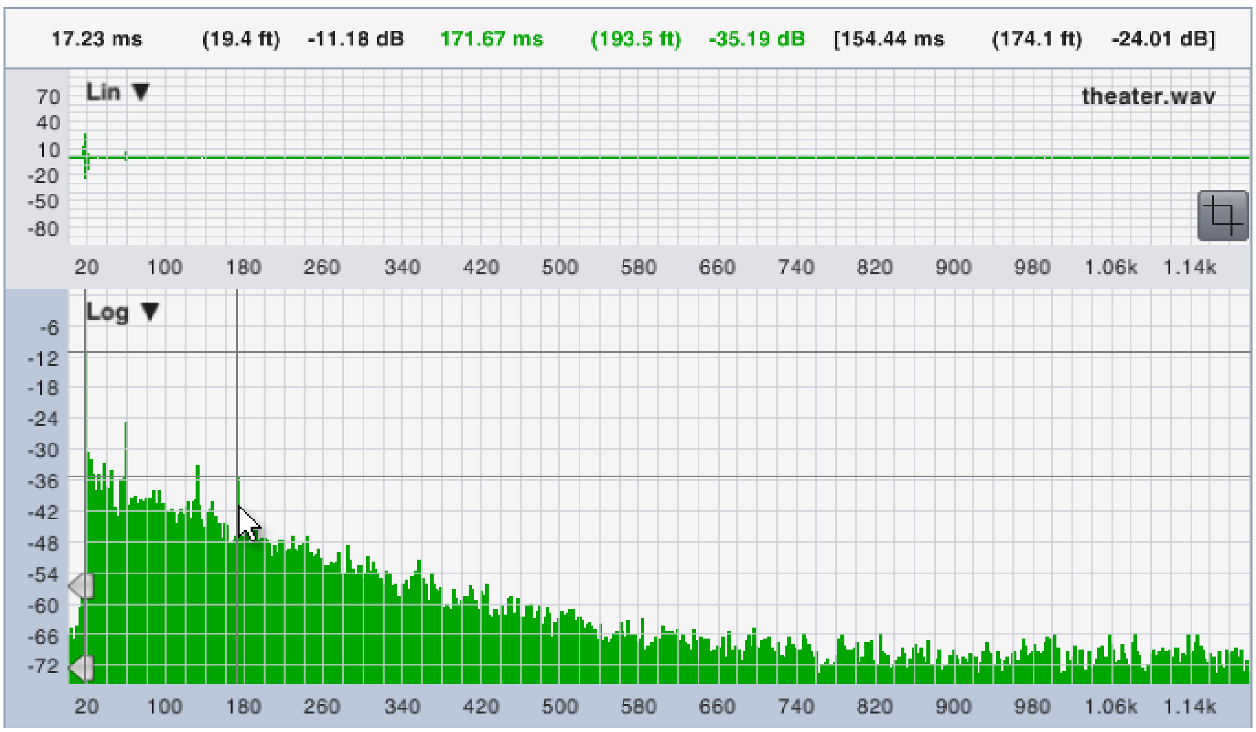


Figure 120: The logarithmic (Log) time domain IR graph plots time on the x axis and magnitude in decibels on the y axis. The combination of locked and movable cursors enables you to find time and level differences between any two points on the plot. Time coordinates can optionally be plotted with equivalent distances as shown above. The pair of coordinates on the far left in the cursor readout at the top of the frame is the locked cursor position, which is set to the highest peak in the IR. The middle pair of coordinates in green is the absolute location for the movable cursor and the rightmost pair in brackets is the difference between the first two.

Most of the examples in this chapter were created using a handful of .wav files that you can download from the Rational Acoustics web site. Wherever applicable, we will tell you which file was used and how to duplicate our settings, so that you can gain a little hands-on experience as we go.

To download the files, visit www.rationalacoustics.com/support/168699-Smaart-v8-Documentation and select *Sample IR Wave Files* from the *Smaart v8 Documentation* section.

Our first example uses *theater.wav*, an IR measurement of a 400-seat historical vaudeville theater. The measurement was taken from the main floor seating area, about 20 feet (6 m) from the stage, using a small horn-loaded PA speaker positioned on the stage lip as the excitation source. If you would like to load the file yourself, open up Smaart, switch to IR mode, then select *Load Impulse Response* from the *File* menu and navigate to wherever the file resides on your hard disk to open it.

Time Domain Analysis

Logarithmic Time Domain Display

The time domain IR display with logarithmic (Log) amplitude scaling is probably the most familiar to anyone much accustomed to looking at acoustical impulse responses. In this view you can find the arrival times of direct sound and early reflections and overlay the reverse time integration of the IR, along with interactive widgets to calculate EDT and reverberation time (on Log and ETC displays only). Smaart provides octave- and 1/3-octave bandpass filters that you can use to filter the IR on the fly, to see how reverberant decay and other characteristics change with frequency.

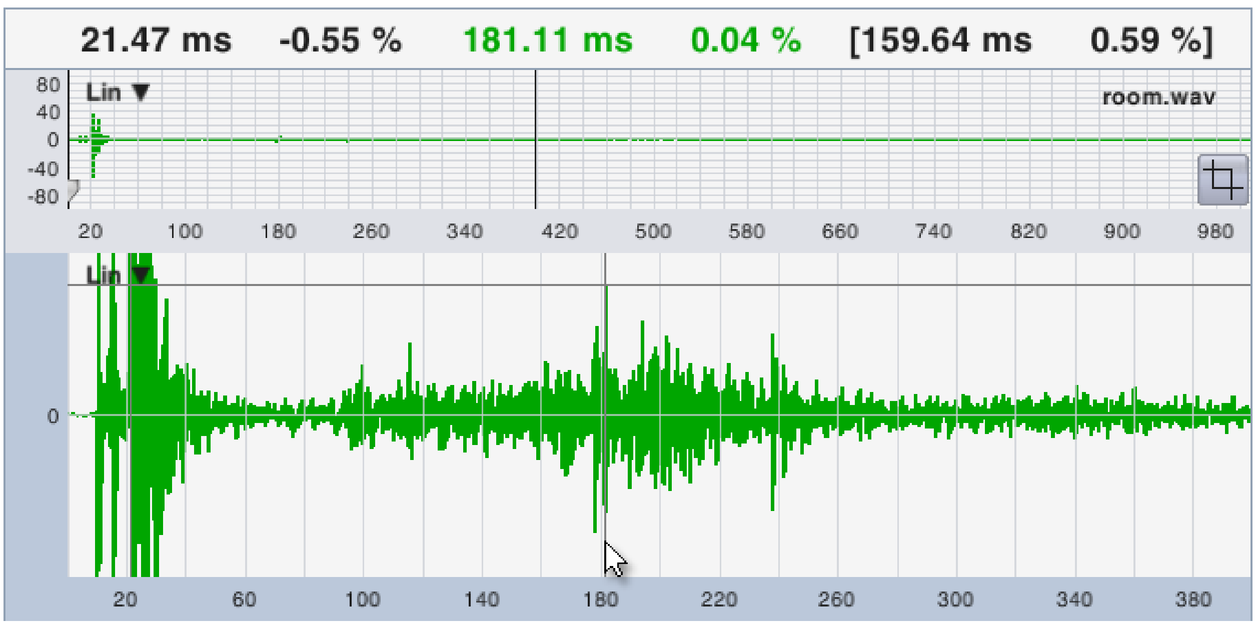


Figure 121: Zooming in on a Linear (Lin) time domain view of room.wav and using the cursor readout to find the relative arrival time of a prominent discrete reflection

The combination of locked and free cursors on time domain displays enables you to find the relative arrival time and amplitude differences between any two points on the plot. The difference between the two is shown in the cursor readout. If the Milliseconds and Distance option is selected in the *Cursor Time Readout* section of the *General* options page (*Options* menu > *General*), Smaart will also give you equivalent distances for time coordinates, based on the current Speed of Sound settings. To move the locked cursor to an arbitrary point on the plot, hold down the Ctrl key (Cmd key on Mac) on your keyboard while clicking with your mouse on a point that you want to mark. Pressing Ctrl/Cmd + “P” resets the locked cursor to the highest peak in the IR.

Linear Time Domain Display

A linear (Lin) time domain chart plots the same data as the Log IR but on a normalized linear amplitude scale, where amplitude values are given as a percentage of digital full scale. This view tends to be of limited usefulness for acoustical analysis in general, however it can be a very good tool for finding discrete reflections, particularly when measuring in an empty hall before an audience arrives. In this case, using the linear IR view can help you to identify hard reflections that might be masked by the diffuse reverberant field on a logarithmic display, only to become much more obvious and audible (often on stage, to the consternation of opera singers) once there is an audience in place and the reverberant levels decrease.

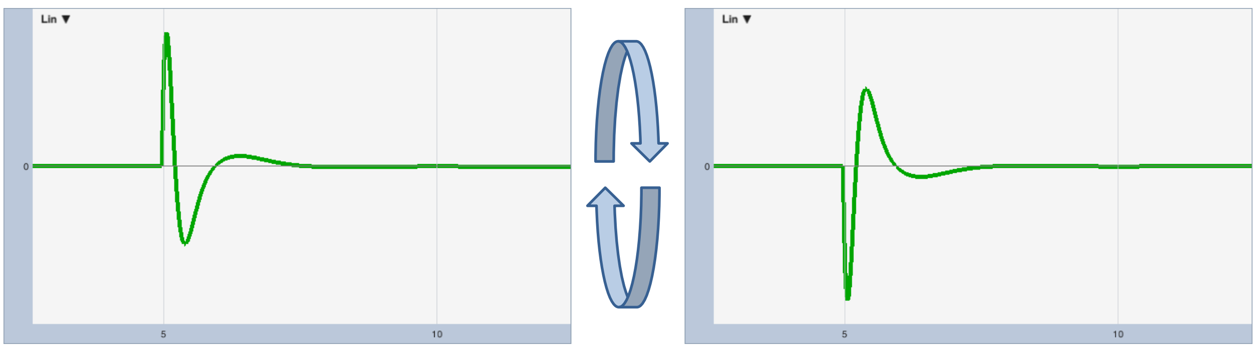


Figure 122: Zoomed in views of the linear impulse response of a bandpass filter, with normal and inverse polarity

Another thing the Linear IR can tell you that the Log and ETC graphs can’t is relative polarity. For example you could measure two midrange drivers or other like devices and determine if they are wired with the same or different polarity by noting which direction the prominent peaks in the impulse are pointed. Figure 122 shows a zoomed in view of the linear (Lin) scaled impulse response of a 2nd order Butterworth bandpass filter with normal and inverse polarity. Cutoff frequencies for the filter are 400 and 1600 Hz. It’s easy to see that the peaks in the two IRs are pointed in different directions relative to each other. Unfortunately, this doesn’t necessarily tell you which one is correct. But if you measured three like devices and one was different, you might reasonably say that the majority rules. Or if you measured two like devices and found opposite polarity and one of them sounded better, it’s possible you might have found the problem. Linear view can also come in handy for looking at other types of signals in the time domain other than impulse responses.

Energy Time Curve (ETC)

The impulse response represents a 2-D graph of a 3-D event: the magnitude and phase of the energy arrival over time. With magnitude on the vertical (y) axis and time on the horizontal (x) axis, phase ends up being represented on the z-axis, which is effectively lost in this view. Consequently, in the linear and log views of the IR, energy arrival that is 90° or 270° shifted shows as a zero crossings, thereby making a single arrival that is spread out over time and phase appear to be multiple arrivals.

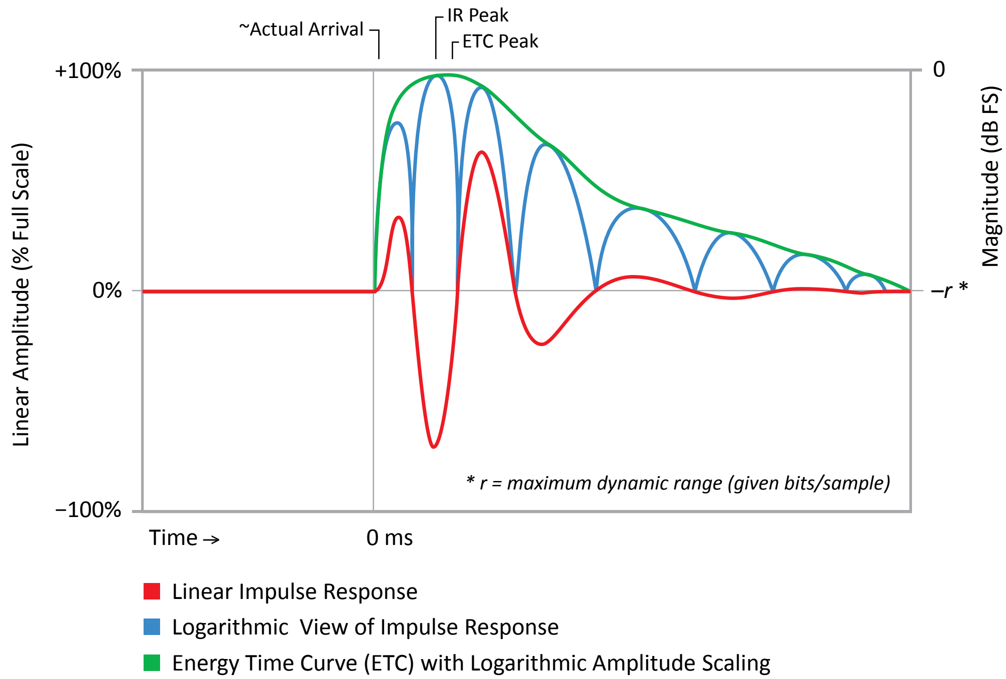


Figure 123: A comparison of the ETC and the impulse response with linear and logarithmic amplitude scaling

The Energy Time Curve, also called envelope of the impulse response, represents the magnitude of the energy arrival over time by effectively ignoring phase. The textbook description is the real impulse response combined with its Hilbert transform – a copy of itself that has been rotated 90° in phase. In practical terms, the summation of the two tends to fill in zero crossings seen in the Log IR, producing a signal that can be a lot easier to look at than the Log IR by virtue of being less squiggly. At higher frequencies the Log IR and ETC may look very similar – both are plotted on a logarithmic magnitude scale – but the ETC is particularly useful for sizing up the arrival of direct sound at low frequencies.

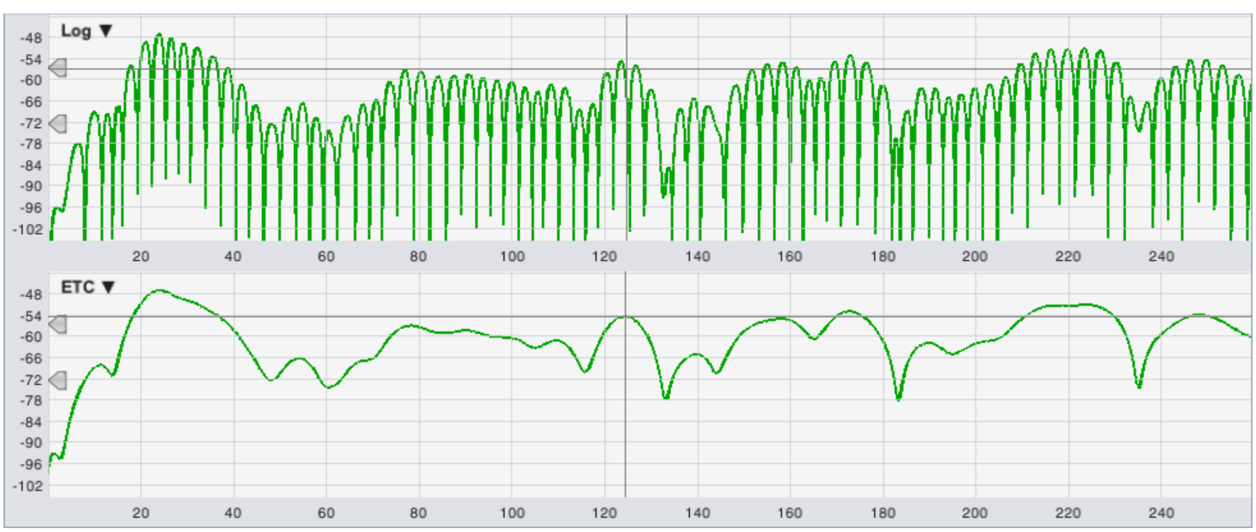
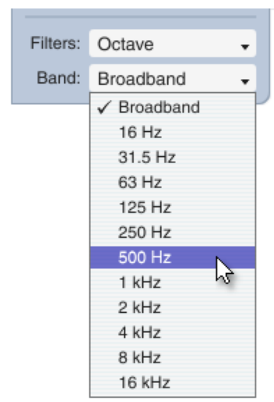


Figure 124: A comparison of the Log IR and ETC graphs in Smaart for the 125 Hz octave band in room.wav

If you zoom in on the first 250 ms of room.wav and switch to the 125 Hz octave band, the difference between the Log IR and ETC is pretty striking (see Figure 124). To scale your display to look like Figure 124, press the plus [+] key on your keyboard a few times to zoom in on the magnitude range then use the up/down arrow keys to move the range up and down.

Note that when using peak locations to find delays, the ETC can sometimes give you a slightly different answer than the Log IR, because of the way it effectively interpolates between peaks in the IR. If you look at the smaller peak in the ETC, at about 124 ms in Figure 124, you can see that it falls in between two lobes in the Log IR. We have found that the ETC can be more effective than the Log IR tool for finding subwoofer delay times. But that is better done in real-time mode, using the ETC on the Live IR in conjunction with the frequency domain transfer function displays, where you can see phase as well as magnitude and watch changes happening in real time as you adjust processor settings

Bandpass Filtering



Up to now we have mainly been looking at the broadband IR, but quite a lot of acoustical analysis is conventionally done using octave, or sometimes 1/3-octave bands, especially as we get into reverberation times and early-to-late energy ratios. Smaart includes complete sets of octave and 1/3-octave bandpass filters for the octaves between 16 Hz and 16 kHz (assuming 48k or higher sampling rate, at lower sample rates you lose some of the upper bands). Bandpass filtering in Smaart is done non-destructively, on demand. To see a filtered version of the IR, select which set of filters to use (Octave or 1/3 Octave) on the Filters selector, then select the band that you want to look at from the Band list.

Smaart's bandpass filters have linear phase response and their magnitude response satisfies the most stringent (Type 0) tolerances for octave and fractional-octave bandpass filters specified in IEC 61260 and ANSI S1.11. If you would like to see the magnitude response of the bandpass filters you can load the wave file *1samplePulse.wav* and bring up the Frequency graph, then step through the *Bands* list to see each filter. Bandpass filtering applies to all main display types except the Histogram chart (which is already filtered into bands). It does not affect the small graph in the navigation pane. Note that filtering the impulse response will clear the Spectrograph display if present and require a recalculation (by clicking the *Calc* button again).

Discrete Reflections

Reflections are a complicated subject because humans are very good at processing them. They may be useful or detrimental, depending on such factors as their arrival time and loudness relative to direct sound (the two biggies), their frequency content and even the angle they arrive from. Discrete reflections can cause audible problems ranging from coloration (timbre change) to image shift to audible echoes, but trying to figure out which reflections are friend or enemy by looking at squiggly lines on a computer screen can be a bit of a dicey prospect.

Short reflections arriving within the first 30 milliseconds or so after the direct sound at relatively high levels are notorious for producing comb filters that muck up our real-time frequency domain analysis;

but humans actually find them beneficial, enhancing the intelligibility of speech and the clarity of music. Outside that early integration window, reflections can still contribute to subjective impressions of presence, warmth, spaciousness, etc. However, the rules are a little different for speech vs music.

Individual broadband reflections arriving at 95 ms or more can destroy speech intelligibility and make life difficult for presenters and performers if they reach the stage. This is the threshold of where strong reflected sounds begin to be heard as separate events (echoes) and can be disorienting for anyone trying to speak or sing. This happens to have been the problem being investigated in the IR measurement shown in in Figure 134 on page 160, where a high-level reflection was arriving at about 160 ms, which is close to the average syllabic rate for normal, conversational speech.

Low-order, early reflections may be visible on time domain plots as individual peaks following the arrival of direct sound. Later arrivals can show up as spikes protruding from the reverberant decay slope. On the Spectrograph plot, higher-level broadband reflections can often be identified as distinct vertical streaks when you run the dynamic range controls up and down, particularly the Max setting. They tend to be most problematic when arriving at longer delay times and relatively high levels, compared to the level of the diffuse reverberant field.

A pretty good rule of thumb is that the later the arrival, the lower in level it needs to be in order to be perceived as beneficial or neutral. Another is that our tolerances for reflected sounds and reverberation tend to be wider for music than speech. Smaart is very useful for identifying problematic reflections; however your ears are probably still the best tool for evaluating their relative significance or severity.

Reverberation Time

Reverberation time (commonly referred to as T60 or RT60, or somewhat less commonly as T30, T20 or simply T) is the time required for reverberant sound energy in a space to decay by 60 dB from an excited level. It is regarded as an important metric in the acoustics of musical performance spaces and also classrooms, auditoriums and cinemas, where it is used as a rough predictor of speech intelligibility.

Reverse Time Integration

Reverberation time is calculated from the reverse integration of an impulse response that has been filtered into octave bands. Conventionally, the 125 Hz to 4 kHz bands are evaluated. Reverse time integration is also called Schroeder integration, after Dr. Manfred Schroeder whose brain child it was. It is a simple thing in concept, but it can be a little tricky to do well.

In theory, you just start at the end of the time record and work your way back to the beginning, tallying up the squares of each sample in the IR as you go. A common problem however, is that the integration will flatten out when the reverberant decay slope runs into the noise floor of the IR. This can lead to overestimation of the reverberation time, particularly if the IR has limited dynamic range, and/or a lengthy noise tail.

The most straightforward solution for this problem is to find the point in the IR where the decay slope meets the noise floor, sometimes referred to as the “saddle point,” and begin the integration there, rather than some arbitrary point such as the end of the recording. The location of the saddle point in an IR is notoriously difficult to estimate automatically though. Smaart 8 uses a proprietary algorithm for IR saddle point estimation that has proven quite robust, but it is not completely foolproof. Therefore, it is still a good idea to check each band to make sure that you agree with the choices the software makes – particularly if there are any large anomalies in the tail of the IR, such as a prominent spike or distortion products piled up at the end of the record by a sweep signal.

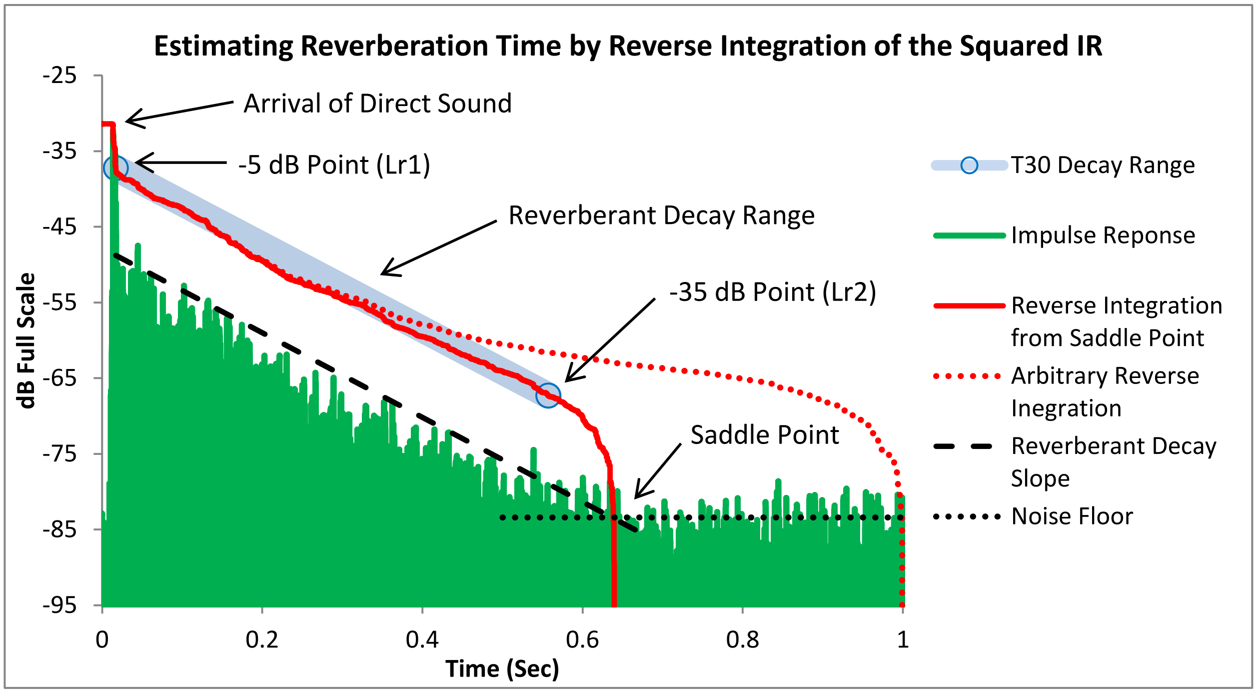


Figure 125: Estimating Reverberation time by reverse integration of the impulse response. Reverse integration of the IR from the "saddle point" – the approximate point where the reverberant decay slope meets the noise floor of the measurement – provides a very good estimation of reverberant decay time. Starting the reverse integration from an arbitrary point such as the end of the file, may result in overestimation of decay time.

Evaluation Ranges (EDT, T20, T30)

Because it is rarely possible to actually measure a full 60 dB of reverberant decay in acoustical systems, reverberation is typically evaluated over a smaller range. The starting point is always 5 dB down on the reverse integration curve from the point corresponding to the arrival of direct sound. The end point of the range is 30 dB down the curve from the starting point, provided that it is at least 10 dB above the noise floor – if not, a 20 dB range may be used. In either case, the measured decay time is extrapolated to the equivalent 60 dB decay time. In ISO 3382 parlance, these are referred to as T20 or T30. Early decay time (EDT) is conventionally measured from the arrival of direct sound down to 10 dB below it on the integration curve. Like reverberation time, EDT is also normalized to 60 dB decay time.

Notice the five level marker widgets shown on the plot in Figure 126. If you were wondering about the cryptic labels, your secret decoder and the default positions for each of the markers is as follows.

- *Ld* = Level Direct. This marker is positioned on the reverse integration curve at the point corresponding to the arrival time of direct sound.
- *Le* = Level Early (Decay). This marker is automatically positioned 10 dB down from the Ld marker on the reverse integration curve. The slope between *Ld* and *Le* is used to calculate EDT.

- *Lr1* = Level Reverberant 1. This marker designates the top of the reverberant decay range, 5 dB down the integration curve from the *Ld* marker. All of the level markers are user adjustable but positioning these three is pretty cut and dried. You *should* rarely find any need to touch them.
- *Lr2* = Level Reverberant 2. This marker designates the end point for the reverberant decay slope. If there is sufficient dynamic range it should be placed 30 dB down the reverse integration curve from *Lr1*. If not, 20 dB will do. *Lr2* is one of the two markers that you may sometimes want to adjust by hand; the other is Ln (below).
- *Ln* = Level of Noise. This is typically the most subjective of the five markers in terms of placement. The time location determines the start point for the reverse time integration curve, which is the basis for positioning all of the other markers. Ideally this will roughly correspond to the saddle point in the impulse response. The magnitude coordinate is used to estimate the level of the noise floor of the measurement and the *Lr2* marker needs to be at least 10 dB above that. Smaart does a pretty good job of placing the *Ln* marker most of the time. However, it may still benefit from a human touch in some cases – particularly if the dynamic range of the measurement is marginal or there are significant distortion artifacts from a swept sine measurement or any other prominent anomalies in the noise tail of the IR being analyzed.

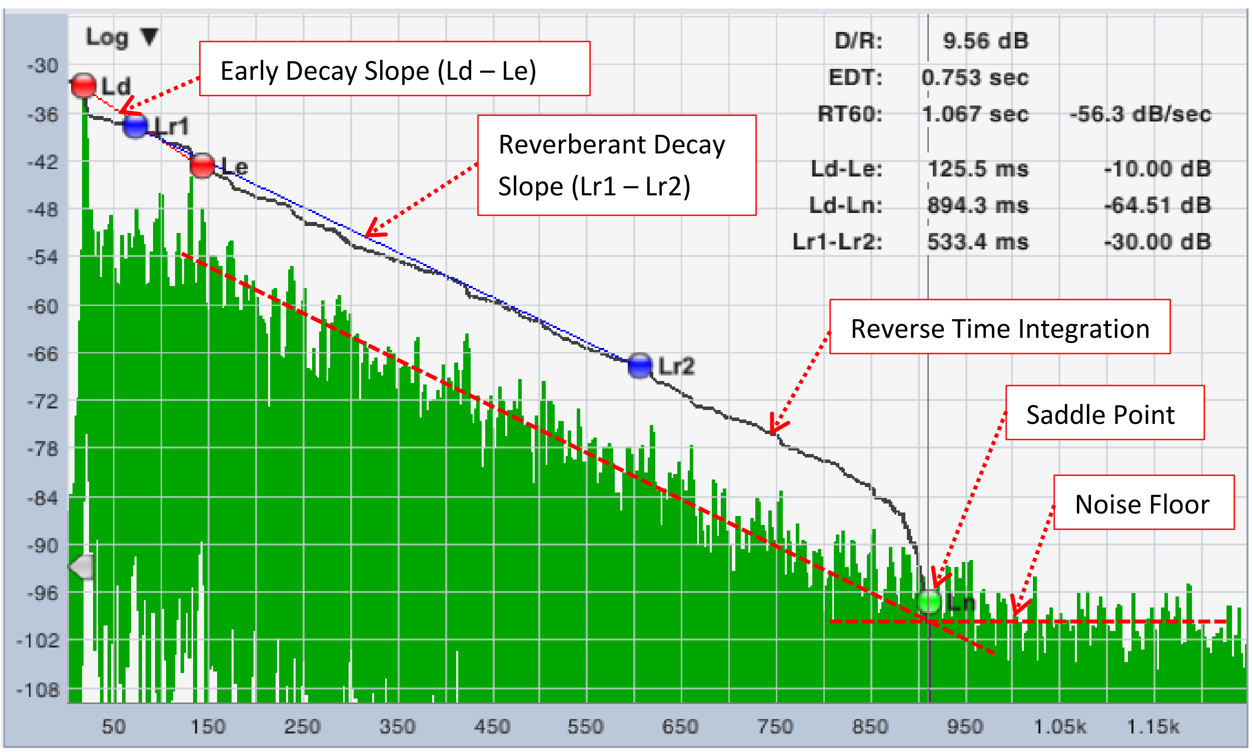


Figure 126: A log IR display with all the bells and whistles. The impulse response shown here is the 500 Hz octave band of theater.wav. Clicking the Schroeder and RT60 buttons displays the reverse time integration curve and the start and end points for the EDT and RT60 evaluation ranges on Log or ETC charts. The positions of all the level markers (Ld, Le, Lr1, Lr2 and Ln) are user adjustable, however they should not require very much adjusting under most circumstances.

When the level marker widgets for reverberation time are visible, a block of vital statistics also appears in the upper right corner of the plot. These include the 60 dB reverberation and early decay times (RT60 and EDT) and the time and level differences between three pairs of markers. The *Ld-Le* level difference should always be 10 dB. *Lr1-Lr2* should be either 20 or 30 dB – this number is convenient for checking your work if you end up adjusting *Lr2* is by hand. The *Ld-Ln* delta is interesting also, as it gives you the dynamic range of the measurement. D/R stands for direct/reverberant ratio. It is an early-to-late energy ratio that gets its split time from the time coordinate of the *Le* marker.

Saving your work

If you do end up adjusting any of the Level marker positions by hand, you can save their positions to a comma-separated values (.csv) text file, by selecting *Save Decay Markers* from the *File* menu. To reload them again later, first load the .wav or .aiff file containing the impulse response, then select *Load Decay Markers* from the *File* menu to open the marker position file.

Reporting Results for Reverberation Time

Reverberation time ideally should be measured from several locations throughout the room and the results from each measurement position averaged together, octave band by octave band to get an average decay time for each octave. Smaart doesn't do that part for you, but the All Bands table does make it easy to get the data from each measurement into a spreadsheet.

Frequency Ranges

The standard evaluation range for reverberation time is the six one-octave bands from 125 Hz 4 kHz. Average times for each octave band can be presented in a table or on a graph. When presenting reverb times on a graph, the frequency axis of the graph should be labeled with the IEC standard nominal octave band center frequencies. The *y*-axis of the graph should have an origin of 0 and be labeled in seconds. It should be noted both in the table and on the graph whether T20 or T30 was used. ISO 3382-1 specifies that if a graph is presented it should be a line graph with a standardized aspect ratio of 2.5 cm per second and 1.5 cm per octave. ISO 3382-2 isn't so picky. It just says "a graph."

Reverberation times for the 125 and 250 Hz bands may be averaged together to get a T_{Low} figure. The average of the 500 Hz and 1 kHz band is called T_{Mid}. When a single number figure is given for reverberation time, it is assumed to be T_{Mid} unless otherwise stated. Smaart calculates these values for you automatically and displays them in the All Bands Table.

Dividing T_{Low} by T_{Mid} gives you the Bass Ratio. Bass ratio quantifies the "warmth" of sound in a venue and is a particularly important parameter for concert halls. The word "Bass" in this case refers to vocal or instrument bass registers and should not be confused with PA-type sub-bass frequencies. Acceptable values are dependent on expectations. A Bass Ratio of 1.1-1.25 would be regarded as good for fairly reverberant concert halls (RT 60 greater than 1.8 seconds) but the upper figure could be increased to 1.45 for less reverberant spaces.

As for what to look for in reverberation time results, preferred reverberation times vary according to room size and purpose and the type of program material being presented. In general, you would like to see shorter reverberation times for auditoriums, classrooms, theaters and cinemas – ideally from about 0.4-0.5 seconds for smaller rooms, up to 0.8 to 1.2 seconds for larger rooms. Opera houses and mixed-use performance spaces where both speech intelligibility and musical appreciation are equally important typically aim for the lower end of the 1.2 to 1.8 second range. Spaces intended for symphonic performances and organ music can range from about 1.8 seconds up to three seconds or more in very large halls.

Reverberation times that are roughly equal across all frequencies are generally preferable for most purposes. The exceptions are things like choral, organ and romantic classical musical music, where a reverberation time curve weighted more toward the lower frequencies may be preferred. It is pretty normal for higher frequencies to decay faster than lows but you don't want to see times that are wildly different in neighboring octaves. In general though, acoustical treatments and/or physical changes to the sound system are typically required to effectively address problems any problems you may find.

Early-to-Late Energy Ratios

Early-to-late energy ratios are another way of objectively characterizing the reverberant characteristics of a room. They are arguably a better measure than reverberation time for any venue where a sound system is an organic part of the acoustical equation. They are simple to calculate automatically and are not subject to the kinds of complications that can make measurement of reverberation times somewhat subjective, but they are a more recent innovation and may be less widely understood than RT60.

Clarity Ratios (C35, C50, C80…)

Clarity indexes are early-to-late energy ratios that compare the integral of the energy arriving within the first *n* milliseconds of the arrival of direct sound (inclusive) to the energy in the remainder of the reverberant decay period. The two most commonly used are C50 and C80, which use at the 50 or 80 milliseconds respectively as their split times. The result of the comparison is expressed as a decibel ratio.

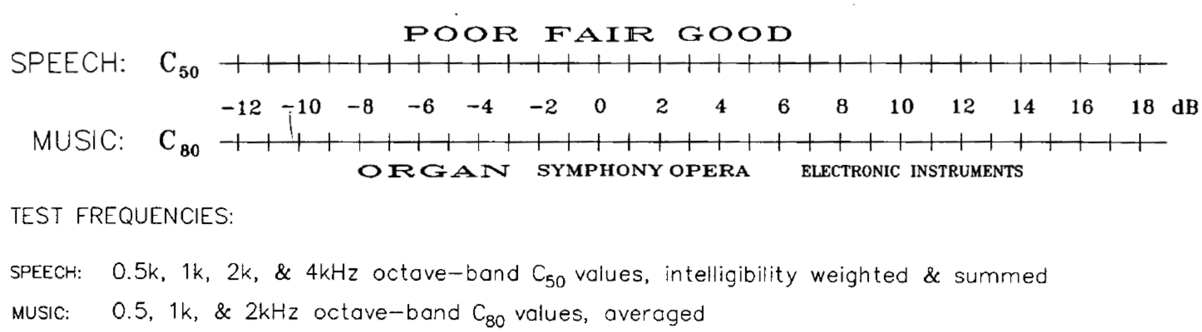


Figure 127: A scale for interpreting C50 and C80 measurement results for speech and music

Shorter split times such as 35 or 50 ms are regarded as better predictors of speech intelligibility. C80 is more useful for music. In terms of what kinds of numbers to look for, Gerald Marshall provided the table shown in Figure 127 in the in a 1996 Journal of the AES article titled, *An Analysis Procedure for Room Acoustics and Sound Amplification Systems Based on the Early-to-Late Sound Energy Ratio.*

For the speech intelligibility side of the graph, Marshall used a weighted average of the 500 Hz to 4 kHz octave bands, with the following weights assigned to each band: 15% for 500 Hz, 25% for 1 kHz, 35% for 2 kHz and 25% for 4 kHz. Others have used the weighting tables for Articulation Index, STI and other scales of their own devising with similar results. For music, he used a simple average of the 500 Hz, 1 kHz and 2 kHz octave bands. We know of no applicable standards for this metric and it has been suggested extending the frequency ranges that Marshall used an octave higher for speech and two octaves higher for music might be useful, but hopefully this example provides a useful starting point for evaluation.

The Histogram Display

Selecting *Histogram* as your display type for an IR mode graph plots a chart of all reverberation times or early-to-late energy ratios by octave or 1/3 octave bands. The type of data the Histogram displays and the resolution are selected by means of the list control in the upper right of the graph. You can change the Histogram to a line chart by opening the *Impulse Response* options page (*Options* menu > *Impulse Response*) and selecting *Plot as Line* under *Histogram Settings*.

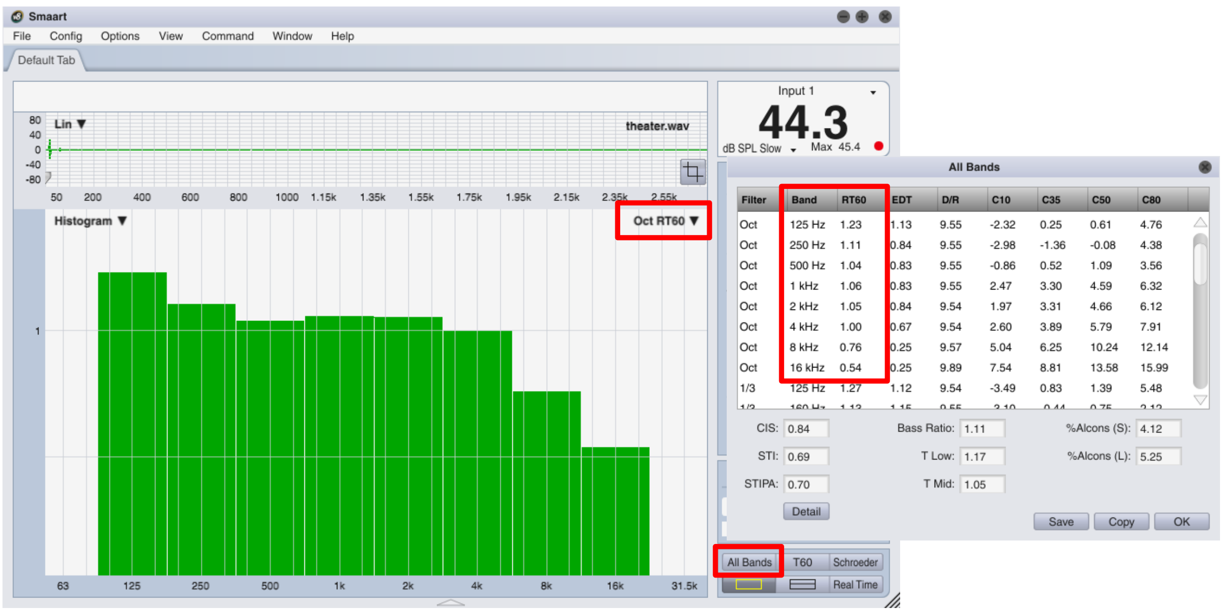


Figure 128: The Histogram graph and All Bands Table. The All Bands Table in Smaart collects reverberation times and early-to-late energy ratios for all octave and 1/3-octave bands in a single table. Speech intelligibility metrics (STI and ALCons) are displayed here as well. The Histogram chart can plot any column of the All Bands Table as a bar graph or line chart in octave or 1/3-octave resolution.

The All Bands Table

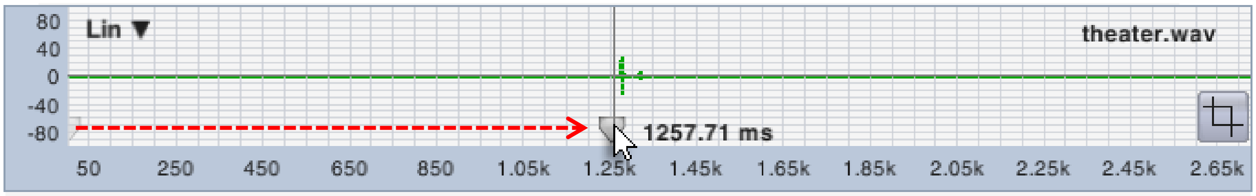
Clicking the *All Bands* button in Impulse mode brings up a report window containing just about every acoustical quantity that Smaart can calculate automatically from an IR, for each octave band and 1/3-octave band where applicable (see Figure 128). Speech intelligibility metrics (STI and ALCons) are displayed here along with *Bass Ratio*, *T Mid* and *T Low*, which are calculated from reverberation times for the 125 Hz to 1 kHz octave bands.

Clicking the *Copy* button in this window copies the entire table to the operating system's clipboard in tab-delimited ASCII format suitable for pasting into a spreadsheet or any other program that accepts ASCII text. You can also save it to a text file by clicking the *Save* button.

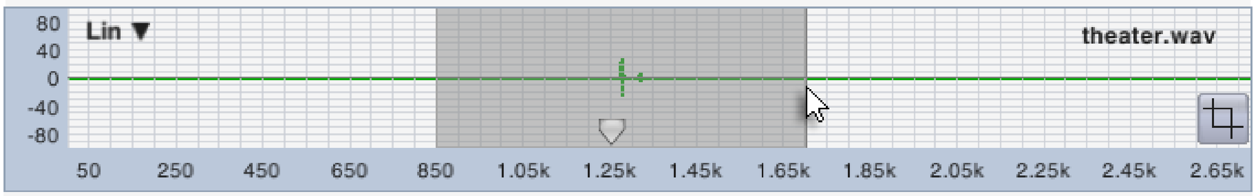
Frequency Domain Analysis

Selecting Frequency as your graph type in the main graph area automatically transforms the IR into the frequency domain to show you its spectrum. The *Frequency* graph has frequency in Hertz on the (horizontal) *x* axis and magnitude in decibels on (vertical) *y* axis. The Smoothing control in the upper right corner of the *Frequency* graph works exactly the same way as smoothing on the real-time transfer function display.

Step 1: Click and drag Time 0 slider to right.



Step 2: Right-click (Ctrl + click on Mac) and drag in navigation pane to select a desired time range.



Result: The spectrum of the selected time range is displayed in the Frequency graph.

Figure 129: Moving the time 0 point and selecting a time range for display. The time range selected in the navigation pane applies to the Frequency graph as well as time domain graphs (Lin, Log or ETC). Note that Smaart uses a tapered data window when transforming any subset of the full IR time range. We've drawn the outline of a Hann window in red on the navigation pane of the "result" portion of the illustration above to help visualize this.

Smaart can calculate arbitrary-length DFTs in IR mode to give you the spectrum of virtually any subset of the IR time record that you care to zoom in on. Time and frequency domain displays are linked so that zooming in on a time-domain graph (Lin, Log or ETC) automatically changes the Frequency display to match. When the entire time record is selected, there's an assumption that you are analyzing a dual-FFT IR measured in Smaart and so no data window is used in calculating the spectrum in that case.

Smaart automatically uses a tapered data window when transforming any subset of the time record, so if you are analyzing an IR file from some other source or a file that has been cropped to less than its original length, you may see better results if you zoom in slightly in the time domain. Tapered data windows significantly attenuate data at the edges of a selected time range – many go all the way to 0 – so you generally want to position any peaks that you want to examine near the center of a selected range. The Time 0 slider in the navigation pane can be used to move peak structures nearer to the center of the time window if they are too close to the edge to center up in the range. Clicking in the right margin of the navigation clears a time zoom.

If you zoom way in on the IR and select a very narrow time range centered on the arrival of direct sound it's possible to see the magnitude response of loudspeaker without comb filters caused by early reflections, at least at high frequencies. In practice, the usefulness of this strategy may be limited to how far away both the loudspeaker and microphone are from the nearest reflecting surfaces. The frequency response of a DFT is limited by its time constant, so you may find that by the time you squeeze the time window in enough to get rid of first order reflections, you can't really see much detail in the frequency domain. But it's something that people used to do quite often in the days before lab-measured anechoic response data for most professional loudspeakers became commonly available.

The Spectrograph

The Spectrograph display in impulse response mode is essentially the same display as the real-time spectrograph. If you understand one, then you understand the other – and if you don't understand either you may want to review *Spectrograph Basics* on page 95. The principle difference between the two is that the IR mode spectrograph is rotated 90° relative to the real-time version, to put time on the *x* axis instead of frequency. In real-time mode in Smaart we want to relate the spectrograph to other frequency-domain graphs, but in IR mode, we most often want to look at it in the context of other time-domain graphs. The other main difference is that the number of "slices" in the IR mode spectrograph is determined by FFT size and Overlap parameters that you select.

To bring up the spectrograph in IR mode, click on the graph selector in the upper left corner of a main graph area pane and select Spectrograph. Initially, you are presented with a blank chart area until you click the *Calc* (calculate) button in the upper right corner of the graph pane. The Spectrograph can eat up a lot of graphics resources when it repaints and so we try to paint it only when necessary. Changing the time range selection in the navigation pane does not affect the spectrograph as it does the other time domain (Lin, Log, and ETC) and Frequency graphs in IR mode but moving Time 0, cropping the time record or filtering the IR will clear the spectrograph and require clicking the *Calc* button again. You can resize the spectrograph and move its range using the [+], [−], and arrow keys or right-click-and-drag on the plot to zoom in on a selected range as you can with any other graph in Smaart. Also, as with other graph types in Smaart, clicking in the left margin of the plot after zooming in will clear the zoom and return the plot to its previous x/y range.

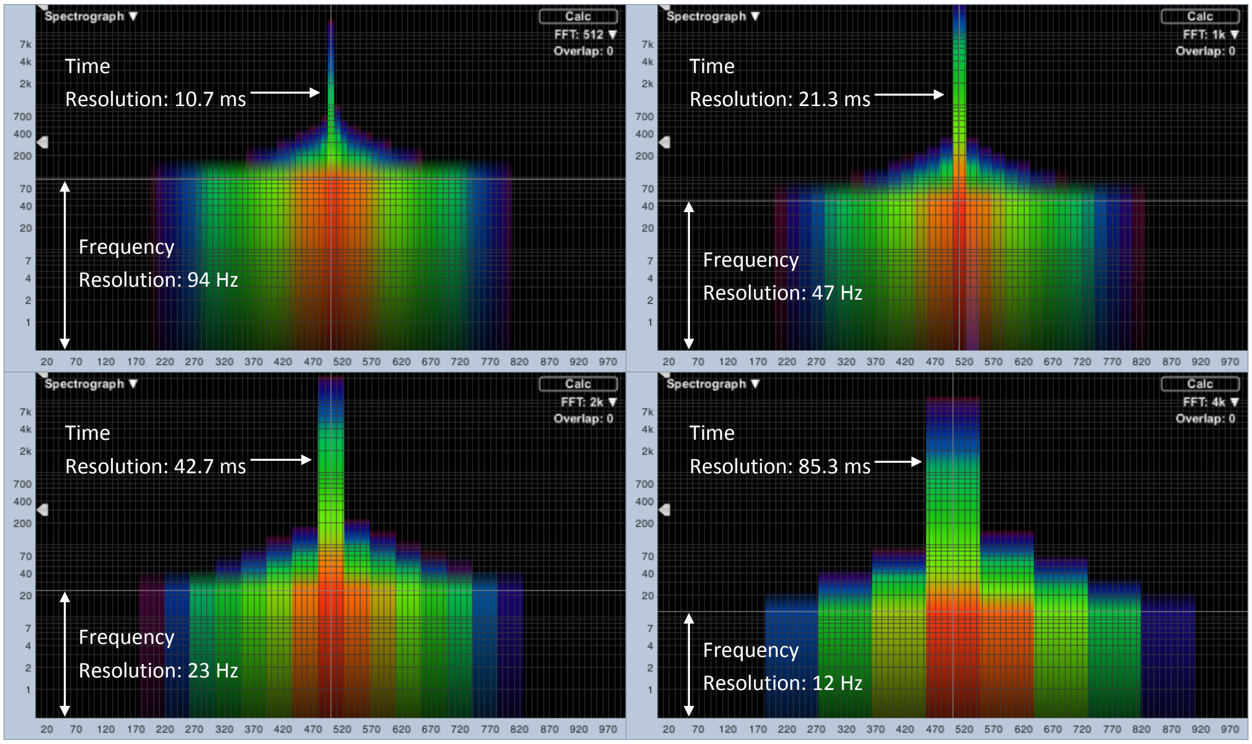
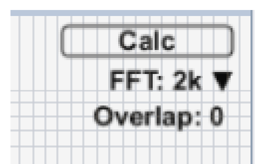


Figure 130: Spectrograph time and frequency resolution as a function of FFT size. FFT sizes ranging from 512 samples to 4K samples are compared, using 0% Overlap. As the FFT size is increased, frequency resolution improved but the peak of the IR is smeared out over a wider time range. The x axis of the graph is time, with frequency on the y axis.

Spectrograph Time and Frequency Resolution



In *Spectrograph Basics* in Chapter 5 (see page 96), we used an octave band analyzer as an example for the sake of simplicity. The IR mode spectrograph is narrowband but the concept is the same. Each vertical stripe in the IR mode spectrograph represents one FFT, meaning the FFT size determines both the time and frequency resolution of the display. Larger FFTs provide greater detail on the frequency axis but may mask transient events on the time axis, so it's a trade-off in that respect. The *FFT* and *Overlap* controls that appear below the *Calc* button determine the time and frequency resolution of the spectrograph display.

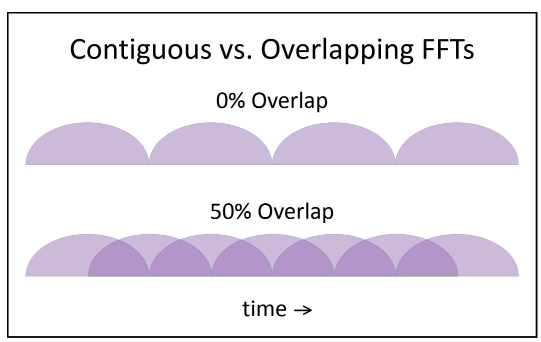


Figure 131: FFT overlap. When set to 0%, each FFT is calculated from unique data. At 50% overlap, the darker shaded areas are shared by successive FFTs. Our "FFTs" are drawn as flattened half circles to suggest a tapered FFT data window.

Figure 130 illustrates how this relationship works. It was created using the file 6dbOctImpulse.wav; the impulse response of a linear phase lowpass filter with 6 dB per octave roll-off. The sharpest part of the peak in the impulse response, where most of the HF energy lives, is probably no more than a few milliseconds wide but notice how its energy is spread across the full FFT time constant in every case. At 512 points, time resolution (the FFT time constant) is a respectable 10.7 milliseconds but the FFT frequency bins are spaced almost 100 Hz apart. Increasing the FFT size to 4K points gets you 12 Hz frequency resolution but smears the peak in the IR over an 85 ms time range.

The other factor affecting the time resolution of the spectrograph is the Overlap percentage that you specify in the upper right corner of the graph. When Overlap is set to zero as in Figure 130, each successive FFT “slice” of the Spectrograph is calculated from unique time domain data, each frame beginning where the last one ended. When Overlap is set to any non-zero value, each successive FFT frame shares some percentage of its data in common with the previous frame(s). The FFT time constant is still the FFT time constant but more overlap can sometimes allude to, if not exactly restore some missing detail on the time axis as FFT size is increased, in addition to producing smoother blending between “slices” (see Figure 131).

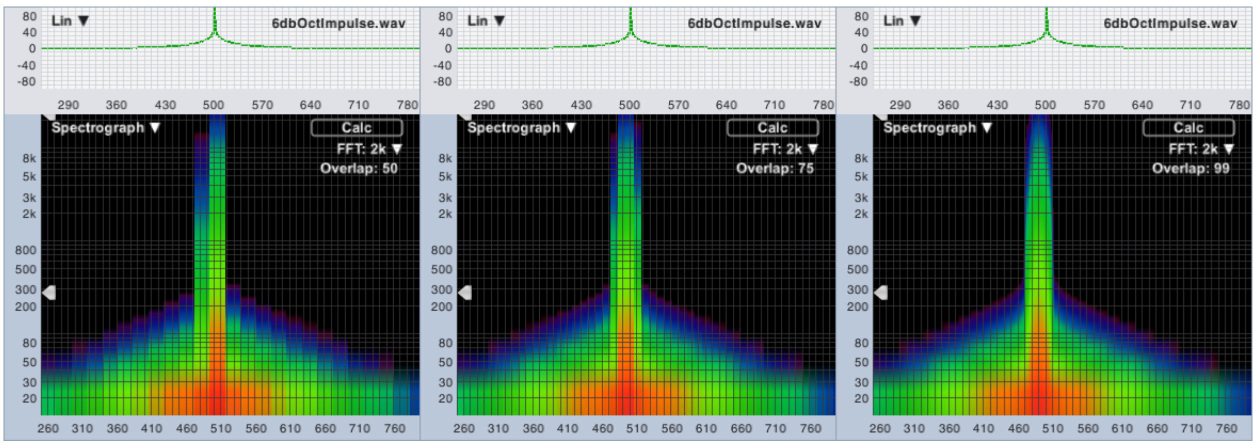


Figure 132: The 2K FFT example from Figure 130, shown with 50%, 75% and 99% overlap(left to right)

Spectrograph Dynamic Range

The dynamic range of the Spectrograph is controlled by two arrowhead-shaped widgets that appear on the left edge of the Spectrograph chart. These controls are echoed on the Log IR and ETC displays where they relate directly to the decibel levels on the graph. The upper of the two widgets sets the maximum (Max) threshold; the lower one sets the minimum (Min). The Spectrograph scales its color spectrum between these two extremes. Any FFT bin whose magnitude exceeds the specified maximum is mapped to the color white. Values falling below the minimum are mapped to black.

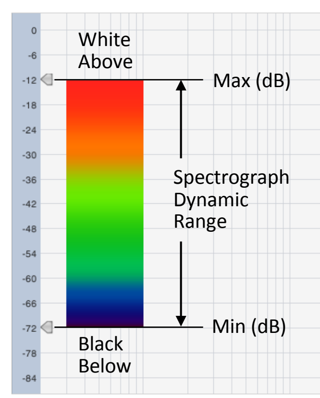


Figure 133: Spectrograph dynamic range and color mapping

Spectrograph Analysis of an Acoustical Impulse Response

Long-time Smaart users may recognize the impulse response measurement in Figure 134 as the *room.wav* file that was distributed with early versions of Smaart. The measurement was recorded on the stage of a 6000 seat performance space using an overhead PA cluster as the excitation source. It features a very prominent reverberant build-up phase and problematic late reflected energy arriving about 160 ms after the direct sound.

This is a good file to experiment with to see how changing the FFT size, Overlap and dynamic range can reveal different aspects of the IR. You can see that we've set the navigation pane graph type to ETC and moved Time 0 to about 100 ms. The FFT size is 2K, overlap is 95% and dynamic range is -20 to -60 dB.

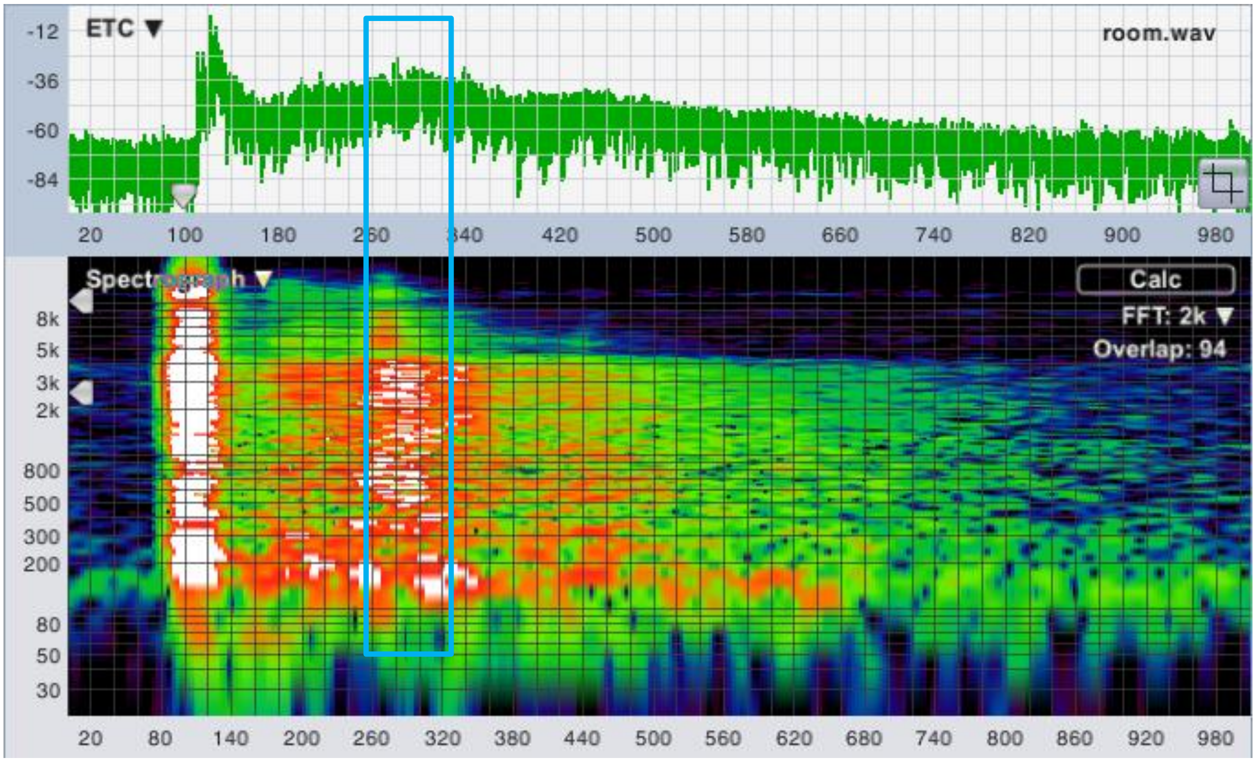


Figure 134: Broadband ETC and Spectrograph of a room impulse response showing a problematic back wall reflection. The spectrograph can be extremely useful for examining both the level and frequency content of features in the IR such as reverberant build-up and discrete reflections.

Speech Intelligibility Metrics

We have discussed the applicability of early-to-late energy ratios as good predictors of subjective speech intelligibility and RT60 as a somewhat rougher gauge. There also exist some purpose-built metrics created specifically for predicting speech intelligibility. Two of these, ALCons and STI, can be calculated from the impulse response of an acoustical system.

STI and STIPA

The Speech Transmission Index (STI) has emerged over the last decade or so as the go-to metric for objective estimation of speech intelligibility in acoustics. STI is a relative of the Articulation Index (AI), which is based on speech-to-noise ratios across a wide range of frequencies. But unlike AI (and its successor, SII), STI works well for reverberant environments in addition to communication systems.

Rather than estimating intelligibility based on direct-to-reverberant or signal-to-noise ratios, STI starts with the concept of speech as a carrier wave (sound from our vocal cords) that is modulated by very low frequency fluctuations as the speaker's mouth and tongue move and change shape to form words (or more precisely, the phonemes from which spoken words are constructed). Looking at Figure 135, a segment of actual human speech, it's not hard to see how someone might arrive at that conclusion.

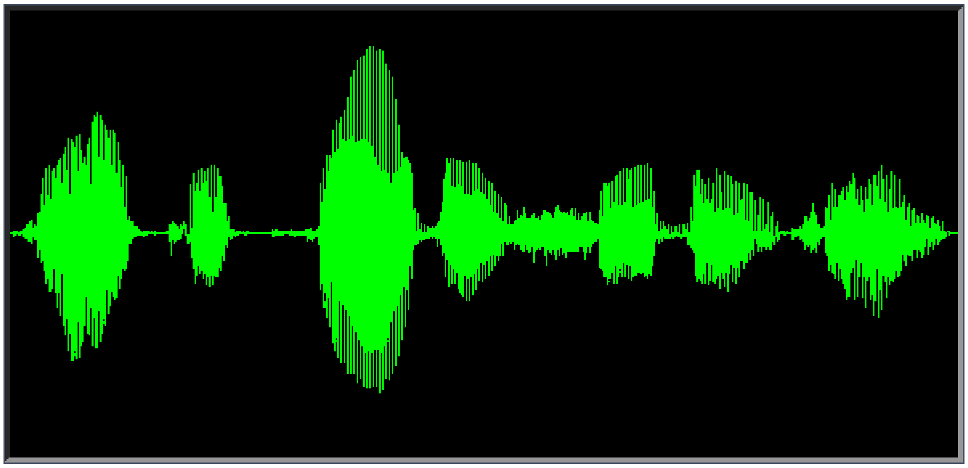


Figure 135: Recording of a male voice saying, “Joe took father’s shoe bench out”

The basic idea is that most of the information in speech is carried in these low-frequency modulations, and anything that reduces the depth of the modulations must negatively affect speech intelligibility. The real advantage of this approach is that STI ends up being sensitive to just about any factor that works to degrade speech intelligibility in a sound system and/or a room, including noise, excessive reverberation, distortion and audible echoes.

The basis for STI is the modulation transfer function (MTF) which quantifies the depth of modulation in the received signal relative to the transmitted signal at specified frequencies. The modulation transfer function can be measured directly using specialized, “speech-like” test signals or calculated indirectly, from the impulse response or ETC of a system under test. In either case, it is measured over a range of seven octaves, from 125 Hz to 8 kHz, at fourteen modulation frequencies per band. The modulation frequencies range from 0.63 Hz to 12.5 Hz in 1/3-octave intervals.

The current IEC standard on STI (60268-16, Edition 4.0 2011-06) says that when measuring STI indirectly from an impulse response, *“The duration of the impulse response shall not be less than half the reverberation time and at least 1,6 s to ensure a reliable calculation of the modulation indices for the lowest modulation frequency of 0,63 Hz.”* This, however, seems to overlook the fact that frequency bins in a DFT are linearly spaced, meaning that the two lowest frequency bins are always an octave apart, whereas the modulation frequencies for STI are on 1/3-octave intervals. If the DFT time window were 1.6 seconds then the two lowest bins would be at 0.63 and 1.25 Hz, whereas the first three STI modulation frequencies are at 0.63, 0.8 and 1.0 Hz.

This is to say that hitting all of the lowest STI modulation frequencies would require a measurement window significantly longer than 1.6 seconds. We experimented and found that a measurement time constant of 5 seconds produces data points very close to all of the STI modulation frequencies. That is the reason for the 5000 ms DFT size for IR measurements. (The exact DFT size in samples depends on the selected sampling rate, for example at 48k sample rate, 5000 ms works out to 240,000 samples.)

IEC 60268-16 qualifies IR-based STI measurement techniques for “noise-free” measurements, where a minimum signal-to-noise ratio of 20 dB in all seven octave bands is obtainable. The standard specifically qualifies MLS and swept sine test signals for used with indirect measurement techniques but also states that, "Theoretically, other mathematically deterministic pseudo-noise (random phase) signals could be employed." Period matched pseudorandom noise in Smaart fits that description and works well for STI.

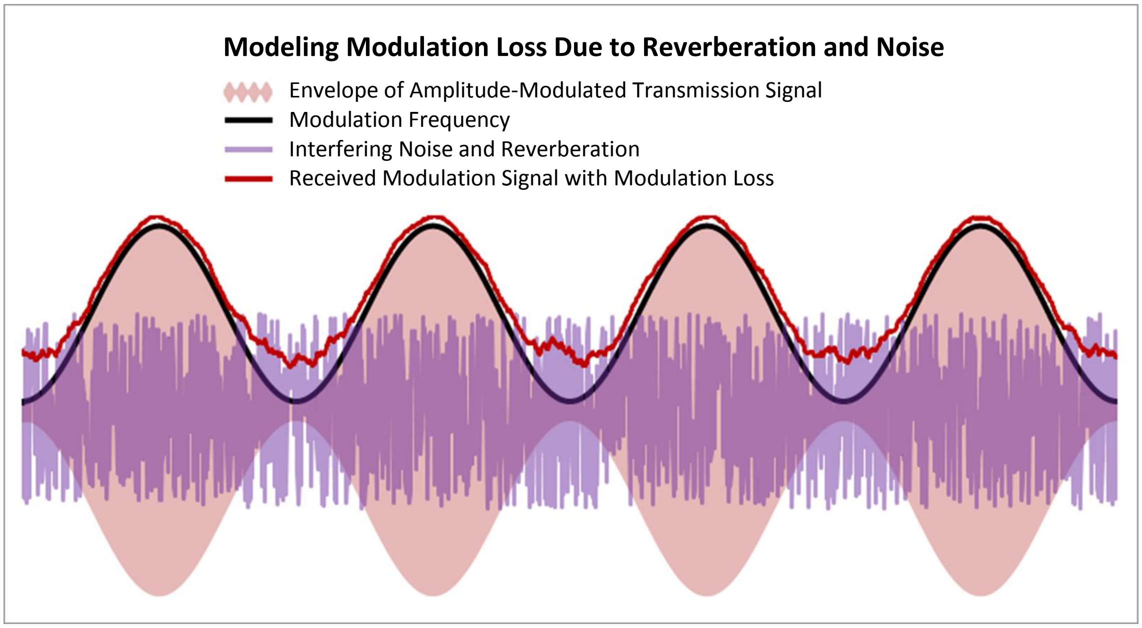


Figure 136: STI estimates speech intelligibility through a transmission channel as a function of modulation loss. The black line in the figure above is the modulation signal for a transmitted signal. The red line is the modulation signal of the received transmission. The difference between the two (the modulation transfer function or MTF), quantifies loss of modulation due to factors such as reverberation and noise.

There are a couple of potential advantages to using period-matched noise over sweeps. One is that it's important to conduct STI measurements at sound levels that reflect actual use of the system under test, and the absolute level of sweep signals that start and stop can be ambiguous. Another is the issue of distortion products potentially piling up at the end of the time record FFT-based IR measurements made with logarithmic sweep signals. Those need to be dealt with in some fashion, either cropped off or windowed out, lest they corrupt the measurement. That isn't an issue with period matched noise although you still need to take care not to overdrive the SUT.

General caveats to using STI are that it is sensitive to strongly fluctuating background noise levels, which can lead to overestimation of low intelligibility systems or underestimation of scores on the high end. When measuring in the presence of fluctuating background noise, at least three measurements should be taken and their results averaged to reduce measurement uncertainty. Also, if the speech source and some prominent source of interfering background noise are widely separated, STI may underestimate intelligibility – human hearing can be smarter than machines about that kind of thing. STI is also sensitive to clipping or amplitude compression in the transmission channel, but in our case, those would also violate the linear time-invariant system rule for transfer function measurements. So don't do that.

Qualitative Thresholds for STI and STIPA

In terms of specific levels to look for, anything better than 0.80 STI is considered excellent. STI scores between 0.61 and 0.80 are "Good," 0.45-0.60 is "Fair," 0.35 to 0.44 is "Poor" and anything less than 0.35

is atrocious. Note that there are male and female versions of STI in the more recent versions of the standard. Any time an STI number is stated without specifying whether it's for a male or female speaker, the assumption is that it's the male version.

STIPA

One of the problems with direct measurement of STI is that it takes a lot of time to make a measurement. The modulation frequencies are so closely spaced that each one had to be measured separately and there are 98 modulation frequencies in all (14 x 7). The full direct measurement takes about 15 minutes to perform as a result. STI for Public Address systems (STIPA) was developed to get around this problem.

STIPA is essentially the same measurement as STI, but uses a subset of its modulation frequencies; two per octave, for a total of 14. STIPA is typically measured directly, using a special test signal that excites all 14 frequencies at the same time, so that the measurement can be completed in a single pass. STIPA measurements can be completed in a few seconds and have been found to correlate very well with the more rigorous, full STI. STIPA is currently validated only for male speakers.

In Smaart, of course, we measure STI indirectly from the impulse response and the full STI measurement takes no longer to perform than a typical direct measurement of STIPA. Smaart does provide figures for both STI and STIPA, however STIPA in our case is more properly termed STIPA(IR), since it is based on IR data rather than measured directly. It is provided for informational purposes, e.g., to facilitate comparison with readings from hand-held STIPA meters and is literally just a subset of the full STI measurement, calculated from exactly the same measurement data.

ALCons

ALCons, sometimes called %ALCons because it is stated as a percentage, stands for Articulation Loss of Consonants. Consonant sounds are critical to speech intelligibility because they are short in duration and tend to get lost more easily than vowel sounds that are voiced over a longer period of time and have more total energy. ALCons was originally conceived as an estimate based on distance of the listener from a sound source, room volume and reverberation time of the room. This is commonly called the "architectural" form of ALCons, due to its reliance on room dimensions and distance. Later forms of the calculation using direct-to-reverberant ratio in place of volume and distance made ALCons suitable for direct measurement from an impulse response and found their way into a number of acoustical measurement platforms including Smaart.

The later forms use an early-to-late energy ratio with a relatively short split time – typically in the range of 10-15 milliseconds – to estimate direct-to-reverberant ratio. The earlier of the two most commonly used directly measurable forms did not take background noise into account, making it suitable only for cases where noise was not significant factor affecting speech intelligibility. A later version that does account for noise is informally called "long form" ALCons, to differentiate it from the earlier, "short form" calculation. The directly measurable forms of ALCons can be calculated for any frequency range but ALCons is regarded being most meaningful in the octave band centered on 2 kHz, as this is where most of the energy in consonant sounds is found.

Advantages of ALCons include the fact that in its original “architectural” form, it made estimation of speech intelligibility possible for sound system designs not yet installed, in rooms not yet been built, without the aid of acoustic modeling programs not yet available in the 1970s and ‘80s. In its later form, it could be directly measured in existing installations based on ETC or IR data produced by TDS and MLS/FHT-based measurement systems that were prevalent before the computing power needed to calculate FFTs large enough for room acoustics work became widely accessible. The main disadvantages of ALCons are its reliance on assumptions that the sound field is statistically well behaved and without audible echoes and also, in the case of electroacoustical systems, that the system is free of audible distortion that could affect intelligibility.

Qualitative Thresholds for ALCons

ALCons is upside-down relative to other intelligibility metrics. Smaller numbers mean better scores. Anything less than 5% is considered excellent. Between 5-10% is “Good”, 10-15% is rated as “Fair” and anything more than 15% is problematic.

Chapter 10: Measuring an Acoustical Impulse Response

You can make an impulse response measurement in Smaart by clicking a single button. Whether or not that measurement returns any useful information depends on decisions made beforehand. The process for measuring an acoustical impulse response can be summarized as follows:

- Selection of measurement technique and stimulus type.
- Selection of excitation source(s) and position(s).
- Selection of measurement position(s).
- Estimation of the reverberation time and background noise.
- Selection of measurement parameters (measurement length/duration, excitation level)
- Exciting the system and recording results

What are we measuring, and why?

Before you set out to make any acoustical measurement, it is always helpful to define your objectives clearly. The trip back to the site that you save may be your own. Obviously we want to measure the acoustical impulse response of a system under test (SUT) for some reason, but what exactly is the system? Is it a room? Is it a sound system? Is it a combination of a sound system its acoustical environment? What do you want to know about the system? What equipment and measurements will be needed to make sure you get the information you need?

If you want to measure the reverberation time of a room with an installed sound system, are you more interested in the room or the system? Consider that using a directional loudspeaker to excite the space may affect reverberation times in locations that are on axis with the speaker. Consider also that when using different speakers to measure from different points in the room, any significant differences between those speakers will show up in your measurement results. If the room is your target, the course of least resistance might be to bring in an omnidirectional loudspeaker specifically designed for acoustical measurement. On the other hand, if your objective is to measure the performance of a loudspeaker system installed in a room, you might be more concerned with early-to-late energy ratios and speech intelligibility metrics than reverberation time of the room, exclusive of the sound system.

Direct vs Indirect IR measurement

There are two basic ways to measure an impulse response in Smaart; direct or indirect. Or you could say there are three possible methods, because the indirect IR measurement method that Smaart employs can be used as a deterministic or non-deterministic measurement technique. Let’s start with the simplest and easiest to understand – old fashioned direct IR measurement – and work our way up from there. This should not be construed as being in any sort of order of preference. In fact, we’re starting with what is usually the least preferred method, but they all have their selling points.

Direct IR Measurement Using an Impulsive Stimulus

The most intuitive way to measure the impulse response of a system would be to use an impulsive stimulus of some kind and simply record what happens. And in fact, people have been doing just that for decades. The advantages are that you do not need a sound system or even a measurement system to perform the measurement. All you really need is some way to make a loud bang and some way to record it. The main problem with this approach – other than the fact that it doesn't tell you anything about an installed sound system (if applicable) – is the scarcity of really good impulsive stimulus sources.

An ideal impulsive stimulus would be a perfectly instantaneous, perfectly omnidirectional burst of energy, containing equal proportions of energy at all audible frequencies. In the time domain, it would appear as a single vertical spike no more than one sample in width. In the frequency domain, it would produce a perfectly flat magnitude and phase trace. When evaluating the response of your system under test (SUT) to this ideal stimulus, you could then confidently assume that anything you saw in the IR that wasn't an instantaneous spike, or anything in the frequency domain that wasn't a flat line, must be the response of your system.

If you loaded the wave file 1samplePulse.wav to look at the frequency response of Smaart's bandpass filters in the previous chapter, you were actually performing a direct IR measurement on the filters using an ideal impulse. Unfortunately, stimulus signals like that do not exist in the physical world. When we need to measure the impulse response of an acoustical system directly, we end up using stimulus sources that are less than ideal. Blank pistols and balloon pops are common sources. Signal cannon, spark gaps, fireworks and even spot welders have been used. The problem with all of these is that their spectral content is not uniform, their envelopes are not instantaneous, they may not really be as omnidirectional as one might guess, and all of these factors will vary to some extent from one measurement to the next. This introduces uncertainly from the start as to which part of the completed measurement is stimulus and which is response. It also limits the repeatability of test results. For this reason, systems such as Smaart that indirectly infer the response of a system to an ideal impulse have become more the tools of choice these days.

Indirect (Dual Channel) IR Measurement

Indirect impulse response measurements are made using dual-channel measurement techniques that mathematically estimate the response of an SUT using continuous or periodic test signals. Three of the four indirect IR measurement methods we can name require specialized test signals such as sweeps (Time Delay Spectrometry and Direct Convolution) or specialized noise (Hadamard Transform/MLS).

The dual-channel transfer function method that Smaart uses for indirect IR measurement also works best using period matched test signals, but unlike the other three it can also produce very acceptable results using random test signals, provided that both the reference and measurement signals are captured. Transfer function-based IR measurement systems work by calculating the frequency-domain transfer function of a system under test (SUT) from the Fourier Transforms of two signals – the signal going into a system and the output of the system in response to this input – and then transforming the result back into the time domain using an inverse Fourier transform (IFT).

Remember that perfect impulse that we were lamenting didn't exist in the real world for direct IR measurements? Well, that happens to be what you get if you take the IFT of the transfer function of two identical signals. So it follows that when we take the transfer function of a stimulus signal and the SUT's response to it, we theoretically should get something very much like its response to an ideal impulse. And in fact that's pretty much what happens in practice when you use a period-matched excitation signal. When you use this same technique with effectively random signals you also get a lot of extra noise, but repeating the measurement several times and averaging the results generally takes care of that, and Smaart makes this easy to do.

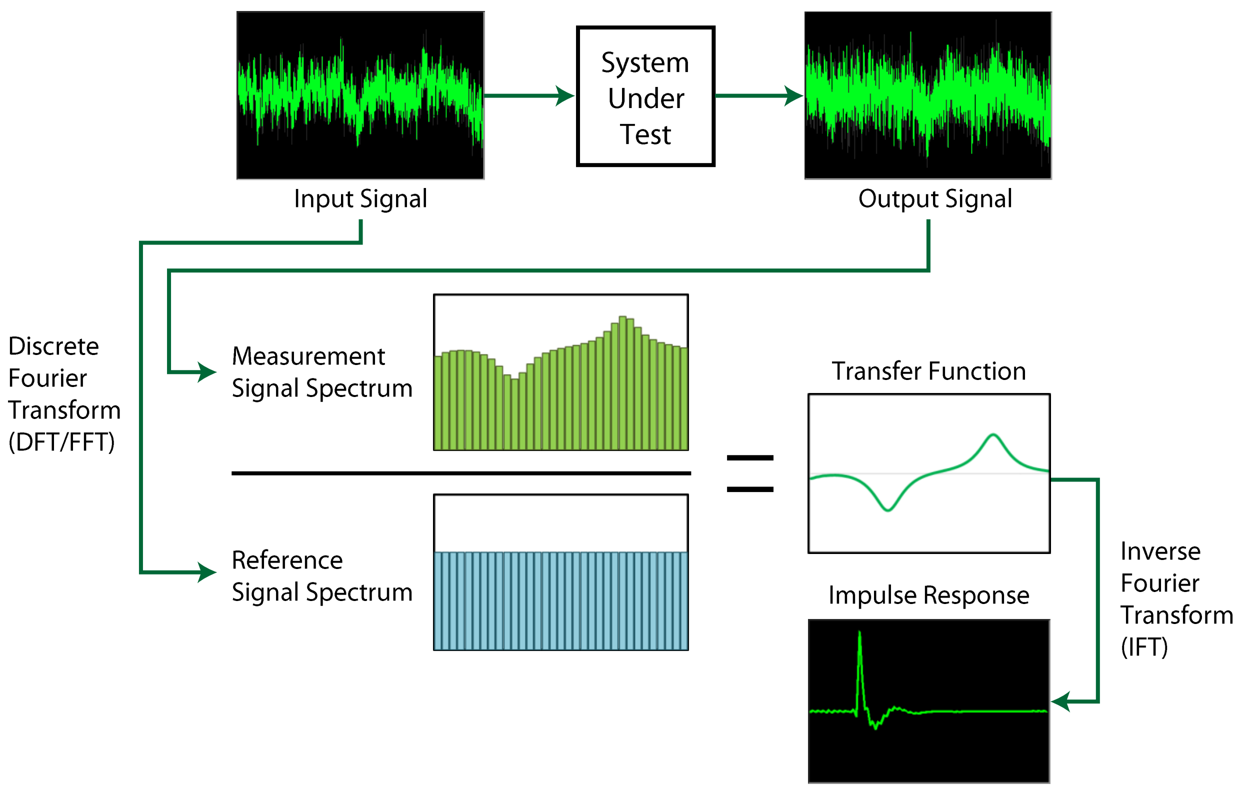


Figure 137: Block diagram of a Dual-FFT transfer function IR measurement

Dual Channel IR Measurement Using Period-Matched Signals

In a way, the discrete Fourier transform (DFT or FFT – all FFTs are DFTs, but not all DFTs are fast) is kind of a dirty mathematical trick. Fourier transforms of all types theoretically work only with signals of infinite length but the DFT gets around this by pretending that a finite chunk of signal being analyzed is really just one instance of an infinitely repeating series of chunks that look exactly like it.

The best way to get around this inherent assumption of this cyclicality in DFT/FFT analysis is to feed the DFT what it really wants to eat: a test signal that either fits completely within the measurement time window or cycles with periodicity equal to the length of the DFT time constant. Signals that meet these criteria can produce deterministic, highly repeatable measurements in a fraction of the time it takes to get comparable results using random signals.

When using matched periodic signals:

- No data window is required.
- Delay compensation is not a critical requirement.
- Considerably less averaging is typically required.
- Measurement time constants can be kept to reasonable lengths.
- Small time variances become less of a concern.
- Subjectivity in selecting measurement parameters is reduced.
- The measurement system doesn't necessarily need to be connected to the system under test. (When using a known test signal, the measurement system and the SUT can get their stimulus/reference signals from two different sources and the measurement will still work. You won't get an accurate propagation delay time without an audio feed from the signal source being used to excite the SUT, but if you don't really need delay times this can be a very handy option.)

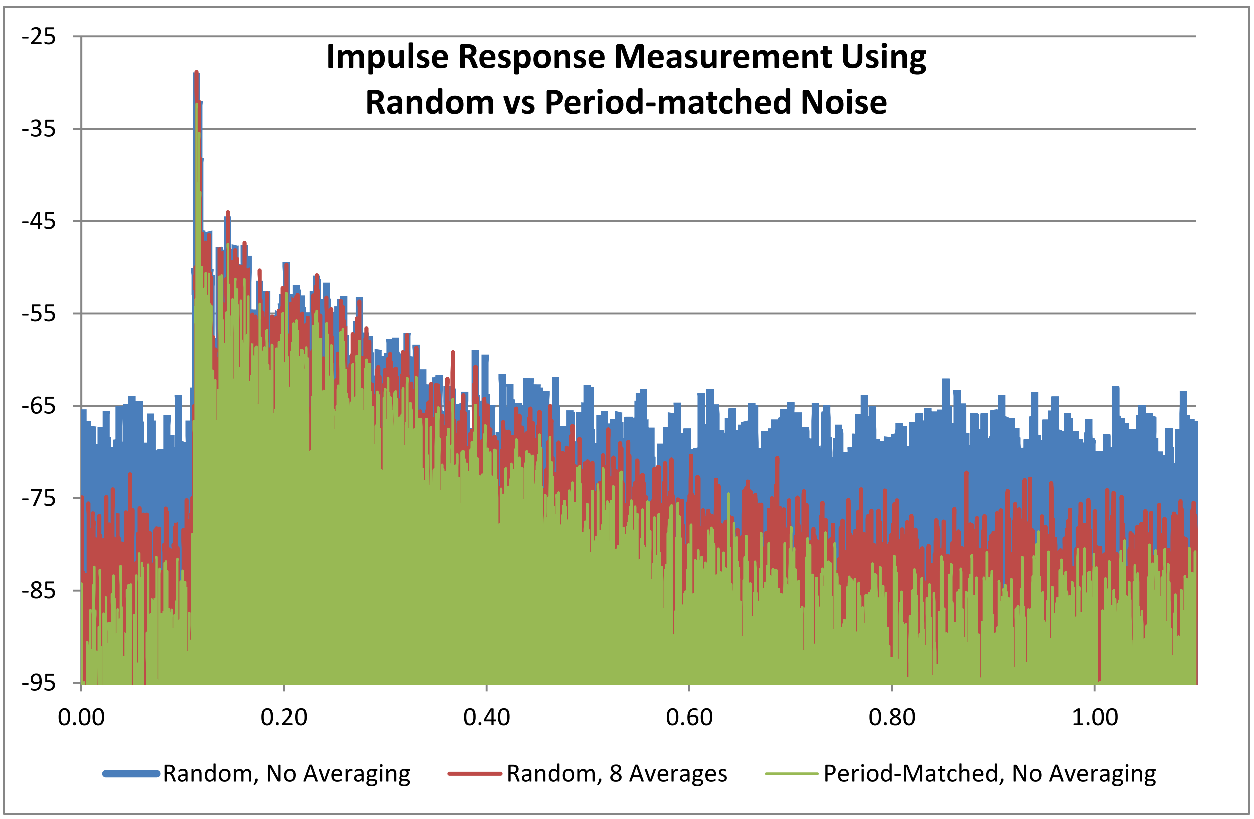


Figure 138: Three indirect IR measurements of the same room, taken from the same microphone position using effectively random noise vs period-matched pseudorandom noise. The period-matched noise measurement (in Green) takes the same amount of time as the unaveraged random measurement (in Blue) but has much better dynamic range. By repeating the random noise measurement eight times and averaging the results (the measurement in Red) we can greatly improve its signal-to-noise ratio, however the measurement takes eight times longer to perform.

Logarithmic Sweeps

Logarithmic sweeps are called Pink Sweeps in Smaart. When you select this signal type in the signal generator, Smaart drops the IR data window without being told. A data window in conjunction with a sweep signal would act as a filter on its frequency content, since each frequency appears at only a single point in time during the measurement.

Sweeps can be used as a circular or aperiodic signal source. If the Triggered by impulse response option is enabled in Smaart's signal generator, the sweep signal is triggered by starting an IR measurement. When you kick off the measurement, Smaart will insert a short period of silence before the sweep in case there's any lag in starting the recording device, then run the sweep and insert another period of silence afterward to let the SUT ring out. If the Triggered by impulse response option is unchecked, the sweep runs continuously when the generator is turned on. In this case you would start the generator before starting the measurement as you would with other test signals.

A peculiarity of dual-FFT-based IR measurements made with logarithmic sweeps is that distortion products in the excitation loudspeaker/SUT are washed out of the IR and show up as "pre-arrivals". Because the DFT is a circular function, these typically end up wrapped around past the beginning of the measurement and pile up near the end of the time record. The practical implication is that you may need to make the measurement time window a little larger than you would for a matched noise measurement, to ensure that these artifacts do not intrude on the reverberant decay slope.

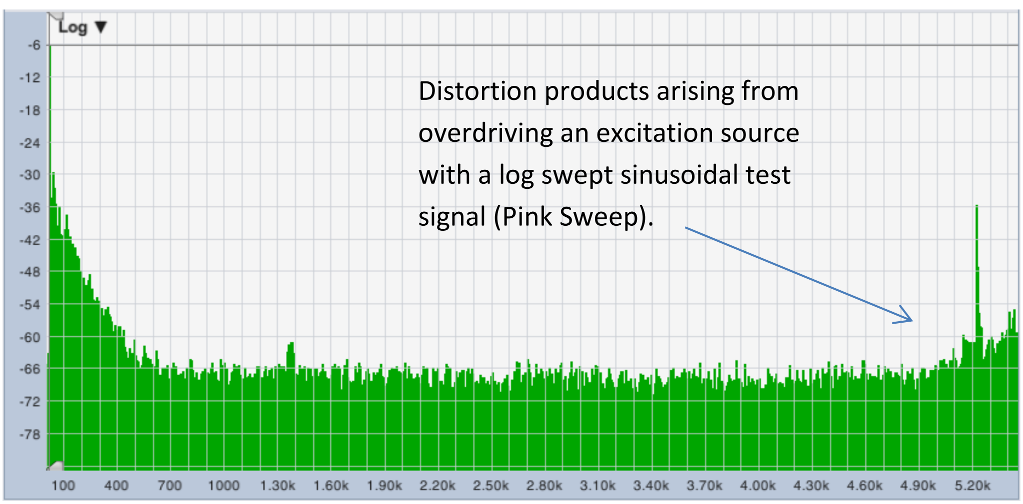


Figure 139: An impulse response measured using a log swept sine signal (Pink Sweep), showing harmonic distortion products from the excitation loudspeaker piled up at the end of the time record. In this case, the speaker being used to excite the room was overdriven and the distortion component was quite significant.

As regards STI measurement, IEC 60628 states that, "When using a sine sweep technique, the distortion components that are inherent within the method shall be edited out or removed from the IR before calculation of the STI can be undertaken." It is our opinion however, that this requirement argues for the use of period-matched noise rather than sweeps for STI measurement, since the masking effect of high levels of distortion in lower-spec announcement systems can significantly affect speech intelligibility, and properly should be included in the measurement.

Dual Channel IR Measurement Using Random Stimulus Signals

An excitation signal that is not completely contained within or, if continuous, has its periodicity precisely matched to the time constant of a discrete Fourier transform being used for analysis is effectively random as far as the DFT is concerned. In Smaart's signal generator, the Random pink noise option or any pseudorandom cycle length with periodicity longer than the FFT size used are effectively random. (Periods shorter than the FFT size should never be used because they won't contain energy at all FFT bin frequencies.) Other examples would include music or noise signals with arbitrary periodicity from sources other than Smaart.

Perhaps the best argument in favor of using random signals for IR measurement is, because you can. If you want to make a measurement with music instead of noise, you can. If it's easier to generate pink noise from a mixing board or in a processor than it would be to inject a test signal into the signal chain from Smaart, that will work. The only absolute requirements are that the measurement system needs to capture an exact copy of the signal going into the SUT and that signal must contain enough energy at all frequencies of interest to you to make a solid measurement.

The main caveats associated with random stimulus signals are a relatively high level of noise, meaning that you have to measure over a longer period than you would with a purpose-built signal to get comparable results. It is left up to the operator to decide how much averaging or how long a time window to use and the actual dynamic range of the SUT is ambiguous. This can be a critical factor in speech intelligibility measurements.

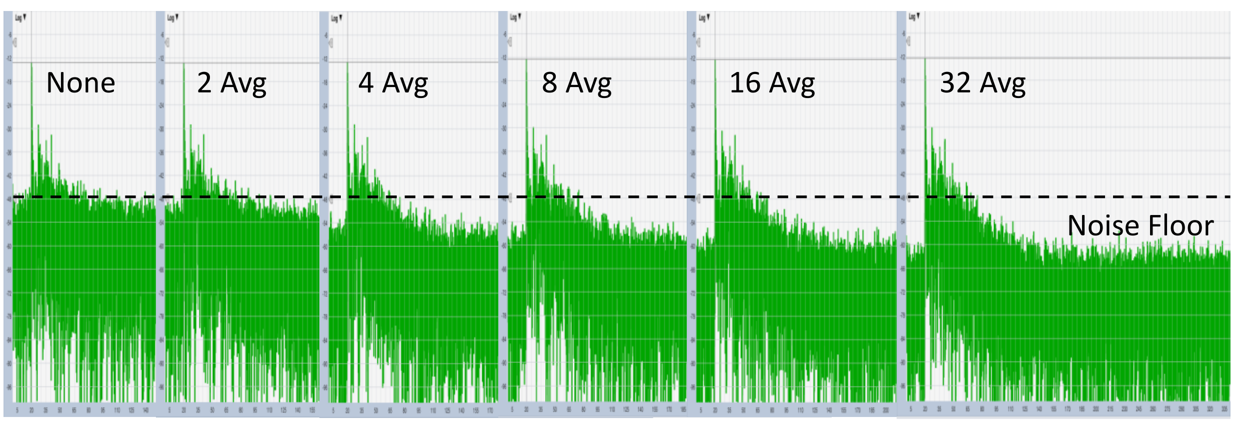


Figure 140: The effect of averaging on an IR measurement made using a random stimulus signal. In theory, each doubling of the number of averages increases signal-to noise ratio by 3 dB.

Reducing Noise in IR Measurements Made Using Random Stimulus Signals

There are three basic things you can do to improve the dynamic range of measurements made using random test signals. The first is to delay the reference signal to match the timing of the measurement signal, so that the data windows line up. You should always do this when measuring with random signals. The second is to evaluate the system over a longer period of time by increasing the DFT size or by averaging multiple measurements (or both). The third is to simply measure louder, which also applies to deterministic and direct IR measurements – in that case you're increasing the signal-to-noise ratio of the measurement by increasing the level of the actual signal, rather than statistically.

Averaging works by inducing regression to the mean in random components of the IR (that is to say, the noisy part). Let's say you take a signal – any signal, maybe an impulse response – and mix it with random noise. Obviously, you get a noisy signal. There is no way to tell just by looking which part was signal and which was noise. But if you take several copies of the same signal and mix each one with different noise, then average all of them together; the noise component of each noisy signal (being random, and different in each case) should start to average toward zero – the theoretical arithmetic mean for random audio noise – while the signal parts (being the same in every case) should average out to themselves.

Of course, all of this depends on the assumption that the signal part of the signal is the same in every case. When working indoors that should generally be a safe assumption. After all, we are working with what we assume to be linear, time-invariant systems in a fairly controlled environment where the worst that could probably happen from one pass to the next is a blast of hot or cold air from an HVAC system causing a slight change to the speed of sound. It might be a larger concern if you needed to make an IR measurement outdoors under windy conditions for some reason. In any scenario where there might be a possibility of any significant time variance during the measurement period, you would probably be better off increasing the measurement time window and/or using a period-matched stimulus signal rather than upping the number of averages.

In theory, averaging two IR measurements or doubling the FFT size used for a single IR measurement should improve signal-to-noise ratio of the measurement by 3 dB. Note that both result in doubling the measurement time, which is really the key to the whole thing. Each additional doubling (2, 4, 8, 16…) of the measurement duration should theoretically get you another 3 dB, although in practice you might reach a point of diminishing returns at some point.

Selecting Excitation Sources and Positions

Excitation source positions should be places that sound would normally emanate from when the system under test is in service. If the loudspeakers you are using to excite the room are the places that sound normally comes from then you've got that part covered. Otherwise an omnidirectional sound source of some kind should be placed on the stage, podium, lectern, pulpit or whatever location(s) that would best simulate normal use of the room/system, and at an appropriate height.

Directional Loudspeakers and Reverberation Time

For the specific purposes of reverberation time measurement, a potential complicating factor may arise if an installed sound system is to be used to excite the room. Impulse response measurements made with directional loudspeakers typically have higher direct to reverberant ratios than IR measurements made by other methods, which could affect reverberation times in the upper octaves. ISO 3382-1 unequivocally states that "the sound source shall be as close to omnidirectional as possible" and provides criteria for assessing the omnidirectionality of a prospective source. ISO 3382-2 specifies measurement procedures for three levels of accuracy in reverberation time measurements: Survey (quick and dirty), Engineering (pretty good) and Precision (very good). For the Precision method, the requirements for the excitation source are identical to those specified in 3382-1 but 3382-2 goes on to say that "For the survey and engineering measurements, there are no specific requirements for the directivity."

Clearly, the spirit of the law is that omnidirectional sources are preferred for reverberation time measurement, but the standard does leave a little wiggle room when measuring in “ordinary rooms” (as opposed to formal musical performance spaces). Given a choice between a measurement made with a directional speaker or not making a measurement at all, a less than ideal measurement is usually better than none. However if any potential errors or subjectivity in evaluating reverberation time are a source of great concern for you, it might be necessary to bring in an omnidirectional measurement speaker and do it by the book – or at least record a few balloon pops, just to have a second opinion.

Reverberation times aside, it is worth mentioning that for most other purposes, IR measurements made using an installed sound system that is actually used for amplified performances in the space you are measuring will be more representative of actual use of the system than measurements made by any other means. In some cases, in order to get everything you need, you might need to make one set of measurements using an omnidirectional source positioned on the stage, another using the installed sound system and perhaps even a third using the house paging system, to estimate its intelligibility.

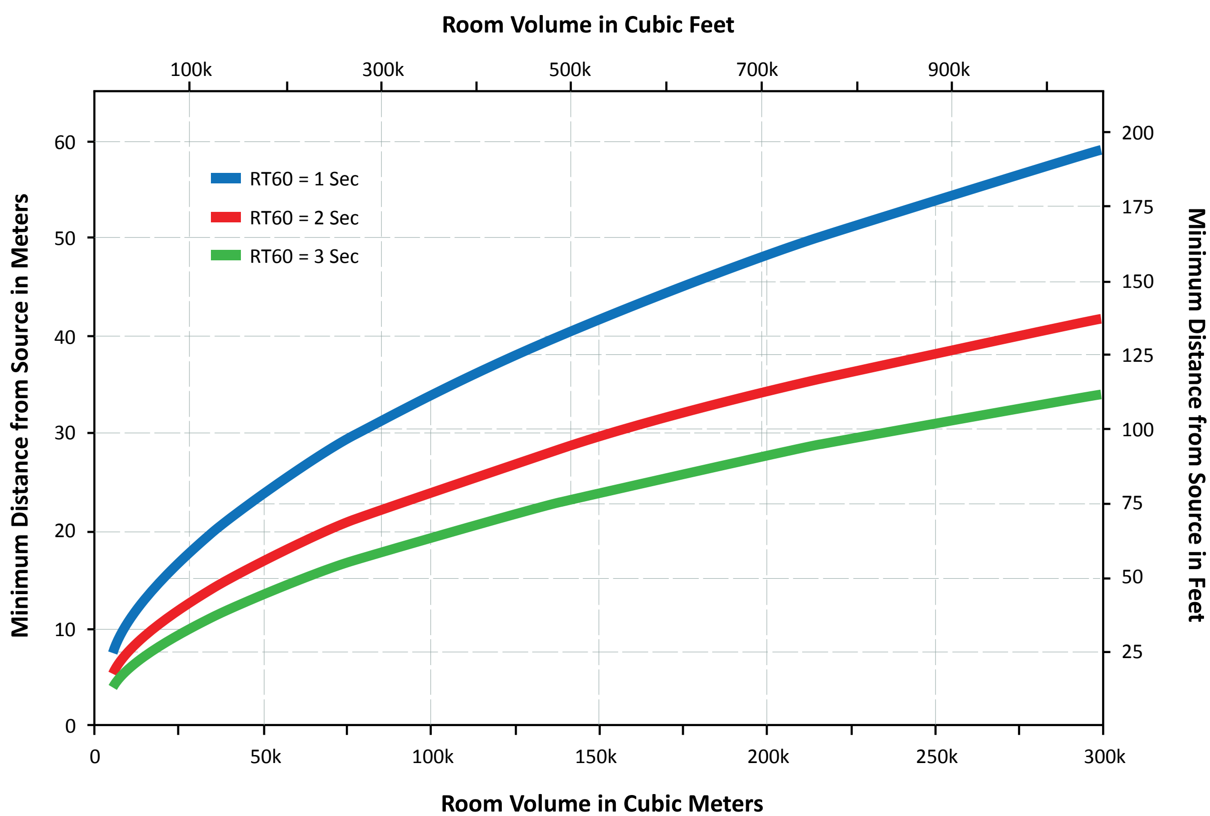


Figure 141: Minimum distance to any measurement position from the excitation source (e.g., a loudspeaker) used for reverberation time measurements. The minimum distance is a function of room volume, estimated reverberation time and the speed of sound, as described by Eq. 1 above. This example uses speed of sound at 20° C (68° F); i.e., 343.6 meters/sec or 1127.4 fps.

Minimum Distance from Sound Sources

Another general requirement for measurement positions used for room IR measurements is that they need to be located far enough away from loudspeakers or other sound sources being used to excite the room to ensure that the measurement is not unduly dominated by direct sound. ISO 3382-2 provides the formula shown in the equation below for calculating the minimum distance (d_{min}) for any measurement position from an excitation source. Figure 141 provides a graph of this relationship.

$$d_{\min} = 2\sqrt{V/_{c\hat{T}}}$$

where

V is the volume of the room in cubic meters
c is the speed of sound in meters/sec
$\hat{T}$ is estimated reverberation time in seconds

Selection of Measurement Positions

The first and most obvious rule for selecting measurement positions is that you generally want to measure from places where one would expect to find listeners when the system under test is in service. If a tree falls in the forest and no one is there to hear it, who really cares if it makes a sound? You might also want to give special attention to any areas where you think there could be problems. Other than that, it is kind of like taking an opinion poll. If we measure from a single position, we have one "opinion" of what the room sounds like. If we sample from a several different locations, we might reasonably expect to see some consensus emerge as to the most common characteristics of the system response, and for position-dependent differences to begin to average out. The more measurement positions, the lower the theoretical margin of error, assuming the positions are chosen so as to be statistically valid.

For the Survey method in ISO 3382-2, a single stimulus source location is measured from at least two measurement locations, providing a theoretical margin of error of ± 10% for octave bands. The Engineering method calls for at least two stimulus source positions and six independent source-microphone combinations for a nominal accuracy ± 5% for octave bands or ± 10% in 1/3-octave bands. The precision method calls for 12 independent source-microphone combinations using at least two different stimulus source locations and reduces measurement uncertainty to no more than ± 2.5% for octave bands and ± 5% for 1/3-octave bands.

ISO 3382-2 specifies that all measurement positions should be at least one-half wavelength apart and at least one-quarter wavelength from any reflecting surface including the floor. For example, if we wanted to measure as low as the 125 Hz octave band, the lower band edge is at ~90 Hz. At 68° F (20° C), the speed of sound in air is 1127.4 feet per second (343.6 mps) and so one wavelength at 90 Hz would be about 12.5 ft (3.8 m). From that, we could conclude that no two mic positions should be less than 6.25 ft (1.9 m) apart and all microphones should be at least 3.13 ft (0.95 m) above the floor and at least that far from any wall or other reflecting surface. For the 63 Hz band you would need to double those distances.

Of course ISO 3382-1 applies specifically to measurement of reverberation time in rooms, so what about acoustical measurements made for other purposes. Two other standards we could look at as a guide to microphone placement are ANSI S1.2, Criteria for Evaluating Room Noise, and SMPTE 202M, the current standard for calibrating cinema sound systems. ANSI S1.2 has this to say about measurement positions:

"Sound measurements for rating room noise under this standard shall be made at locations that are near the average normal standing or seated height of human ears in the space: 5'-6" for standing and 4'-0" for seated adults – 3'-6" standing and 2'-6" for seated children. The microphone shall be no closer than 2'-0" from any sound reflecting surface or 4'-0" from the intersection of two intersecting reflecting surfaces, or 8'-0" from the intersection of three intersecting reflecting surfaces."

SMPTE 202M recommends that microphones be placed:

"In indoor theaters, at position S […] and position R […] should it exist, and at a sufficient number of other positions to reduce the standard deviation of measured position-to-position response to less than 3 dB, which will typically be achieved with four positions. […] It is recommended that measurements be made at a normal seated ear height between 1.0 m and 1.2 m (3.3 ft and 4.0 ft), but not closer than 150 mm (6 in) from the top of a seat, and not closer than 1.5 m (4.9 ft) to any wall and 5.0 m (16.4 ft) from the loudspeaker(s)."

(Position "S" generally works out to be a little left or right of the approximate center of the room on the main floor. Position R is for balconies.)

We can see that these are all in general agreement (depending to some extent on frequency) even though one talking about reverberation time, another is for background noise surveys, and the third is for RTA measurements of cinema sound system. They are probably also not out of line with positions you would intuitively choose for frequency-domain transfer function measurements of a sound system.

Selecting Measurement Parameters

Once you have done all the groundwork of determining your source and measurement positions and figuring out which measurement technique to use, the part of the measurement procedure that directly involves Smaart is actually pretty easy. Basically, you just need to select your measurement parameters, turn on the signal generator (or other stimulus signal source) and kick off the measurement. The two main things you need to concern yourself with at that point are the stimulus level and the measurement duration, which will be some combination of the FFT size and the number of averages.

Input source

If you already have one or more Transfer Function measurements configured and will be using one of those to make your measurement, use the *Group* and *TF Pair* selectors shown in to select the one you want. To create a new TF pair, click the little hammer and wrench button next to the *Group* selector to open the *Measurement Config* window, then click the *New TF Measurement* button (see Figure 142). This pops up another dialog where you can select the input device and channels that you want to use and give the new measurement pair a name. If you are unfamiliar with how to set up your measurement system for transfer function and dual-channel measurements, *Appendix E* has example setup diagrams.

Excitation Level

The rule of thumb for setting the excitation level for IR measurements is that you would like to be able to get at least 40-50 dB above the background noise level. In reverberation time measurements, we evaluate reverberant decay over a range starting 5 dB down from the arrival of direct sound (normally the highest peak in the IR) and extending down another 20 or 30 dB from the start point.

A 30 dB range is preferred but 20 dB is OK if you can't get 30. Either way, the lower end of the range needs to be at least 10 dB above the noise floor of the IR measurement. When you add that all up, you're looking for a minimum of 45 dB of dynamic range for a 30 dB evaluation range, and at least 35 dB for a 20 dB range – and that's in a perfect world, with no noise artifacts from the measurement process itself. In the real world, adding another 5 to 10 dB on top of that would be a definite nice-to-have (unless of course that would drive the system into distortion or blow something up).

To figure out how loud you need to be, you can simply measure the background noise level. We are looking for a relative relationship so you don't even really need to be calibrated for SPL (unless perhaps you plan on doing an STI measurement). Just set the sound level meter in Smaart to *Slow SPL* and watch the meter for ten or twenty seconds with no output signal running to get a feel for the baseline noise level, then start the signal generator at a low level and gradually increase the gain until you reach the target excitation level (or as close to it as you can reasonably get).

Input Levels

Once you have your output levels nailed down, adjust your input levels (by whatever means available) until both the measurement and reference signal levels (labeled M and R on the Control Bar) are roughly even and running at a reasonable level. The yellow segment of the meters in Smaart runs from -12 dB to -6 dB full scale and that's the target zone. The meters are peak reading and we hard limit peaks for noise signals in Smaart, but you have to also allow for fluctuations due to background noise in acoustical measurements and if you use noise from another source you may see wider variations in the peak levels. With sinusoidal sweeps you can run the levels a little higher if you like, due to the lower crest factor of the signal but you always want to keep the levels out of the red. If you are doing a single channel measurement, you will probably have to waste a few balloons or fire off a few blank cartridges while adjusting the measurement channel gain to get a good solid signal level with no clipping on the input level meter.

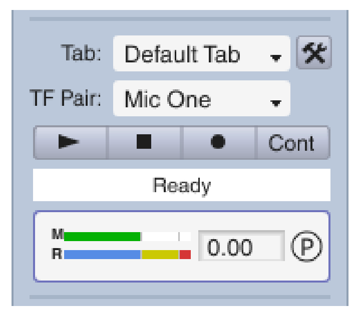


Figure 142: The measurement signal level (M) is running at a comfortable level. The reference (R) channel is clipping

Measurement Duration (Time Window)

For dual-channel IR measurements, the time window is a function of FFT size (see Figure 143). If you are only concerned with measuring delay times then a good rule of thumb for how long the measurement needs to be is 3 times the longest delay time that you want to measure. If you want to measure reverberation time and early to late energy ratios, then the 60 dB decay time (RT60) of the system is a good target. This is kind of a functional requirement for period-matched dual-channel measurements but it's also a pretty good practical target regardless of how you are measuring. Ideally you would like to measure 30 dB of reverberant decay and the lower end of the evaluation range should be at least 10 dB above the noise floor, so that's 40 dB which of course is two thirds of 60. By the time you factor in propagation delay, early decay and maybe enough of a noise tail to see the dynamic range of the measurement, chances are you have eaten well into the remaining third.

Of course, both of these rules require either knowing the delay time or RT60 before you've measured them. That typically means you have to guess, then measure, then possibly adjust your guess and measure again. For delay times you can use the distance to the source divided by the speed of sound as a starting point. For "guesstimating" purposes, you can use 1130 feet or 345 meters per second at typical room temperatures – the speed of sound increases with temperature so if it is very hot where you are working you might adjust your estimate upward a little, or downward if it's cold.

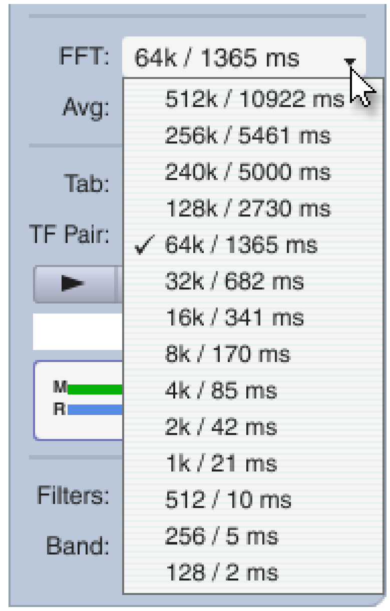


Figure 143: IR Mode FFT size selector showing the time constant in milliseconds for each FFT size

For reverberation times, one to two seconds should at least get you in the ballpark for most theaters and auditoriums. Stadiums and other large structures can have much longer reverb times. There's never any harm in measuring over too long a period, so you may want to err to the high side. If you make a preliminary measurement and you are happy with the results you might even be done. If not, you can adjust accordingly and measure again. Note that as a rule, lower frequencies tend to decay more slowly than highs, meaning that the limiting factor may be the reverberation times in the lowest octaves that your stimulus source can excite. So be sure to check the lower bands when estimating reverb times.

FFT Size (Time window)

For dual-channel measurements the duration of the measurement is determined by the FFT time constant – that is, the time required to record enough samples for a given FFT size at whatever sampling rate you are using. In Impulse mode, Smaart gives you the time constant in milliseconds, along with the frame size in samples for each available FFT/DFT size.

Averaging and Overlap

Averaging, as we discussed earlier in this chapter, is primarily something you concern yourself with when using effectively random stimulus signals. With random or effectively random stimulus signals, deciding how much is averaging is enough is kind of judgment call but typical settings are in the 4-16 range. In very noisy environments, you may want to use a larger value and/or consider using a period matched signal. When measuring with period-matched noise or sweeps, averaging is normally set to "None" or 2, although it is still possible that a higher setting could prove helpful if measuring in an extremely noisy environment.

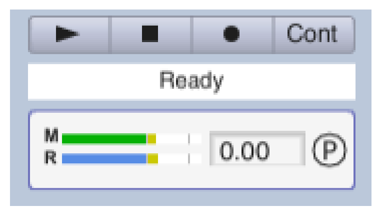
Another factor that affects how averaging works is the Overlap setting found in Impulse Response options (Options menu > Impulse Response). When overlap is set to 0% each FFT is calculated from unique data, giving you the maximum amount of noise reduction that you can get from a given number of averages. When you set the measurement *Overlap* to a non-zero value then successive FFTs share some data in common (see Figure 131 on page 158) – remember that *measurement* overlap and *spectrograph* overlap are two different things, but the principle is the same. If measurement overlap is set to 50%, it only takes a little longer to record 16 averages than it would for 8 at 0% overlap. You don't get the full benefit of averaging 16 unique FFTs in that case and processing time increases but you should see at least a little better signal-to-noise than you would get 8 with some net time savings.

Delay Compensation

When making IR measurements with random signal sources you will get much better results if you compensate for the delay time through the system under test. So plan on making the measurement twice if you don't already know the delay time; once to find the delay and a second time for a keeper. The button labeled "℗" (for Peak) that appears next to the input level meters in IR mode (see Figure 142) sets the reference signal delay to the highest peak in the impulse response.

Pushing the Button and Making the Measurement

Having nailed down which measurement technique will get you the results you need, chosen your excitation sources and measurement positions, set your input and output levels, and selected your FFT length and number of averages (if applicable), all that's left to do is push the button(s). For a dual-channel measurement, start your excitation signal (unless you're using a triggered sweep in Smaart, in which case it will start by itself) and click the start (▶) button in the Control Bar on the right side of the main window. Smaart will take it from there and display the measurement results when it has finished.



For a single-channel (direct IR) measurement, click the record (●) button, then click the start (▶) button, pop your balloon or fire your blank pistol (or whatever), give the system a few seconds to ring out, and then click the stop button (■) to end the recording and display the results.

Saving Your Work

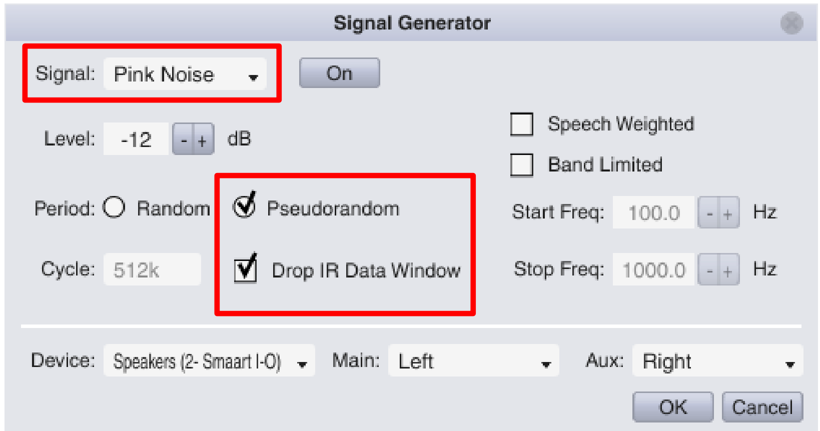
When you measure an IR in Smaart does not automatically assume that you want to save it. Sometimes it takes a few tries to get it right and we don't need a lot of old baggage piling up in the process. When you have a measurement that you are happy with and want to save, click in the File menu and select Save Impulse Response then select the directory where you want to store the file and give it a name. Note that if you have cropped the file for display purposes using the Crop function, only the displayed portion of the time record is written to file.

Recap: Common Settings for Dual-channel IR Measurements

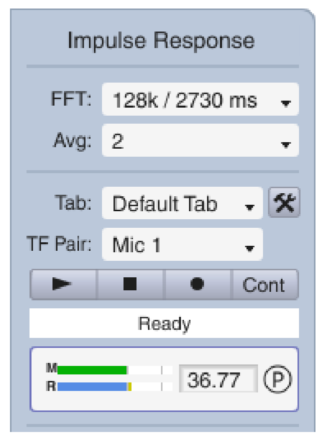
The following are some common "go-to" default settings for IR measurements in Smaart that should typically work well for most rooms.

Signal Type

If you are able to use Smaart's signal generator as your stimulus signal source then period matched pseudorandom noise is a good all-around choice for signal type. To turn on this option, open the signal generator control panel, select Pink Noise as the signal type, then tick the boxes labeled *Pseudorandom* and *Drop IR Data Window*.



FFT size and Averaging



128K makes a good default setting for FFT size. At 48k sampling rate, that gives you almost 3 seconds of time window. You would generally have to be measuring in a pretty huge space to need more than that, but it's not ridiculously long for smaller venues. For averages, a setting of 2 is a good when using period-matched noise. If you have to use a random signal source for some reason, up the number of averages to 8, or maybe 16 if you are working in a noisier environment. We are assuming 0% overlap for averaging.

Excitation Level

If you need to measure reverberation time, then your excitation level needs to be a minimum of 45 dB above the background noise level for T30 (preferred) or at least 35 dB above to get T20. For most other purposes, any excitation level that is comfortably above the background level should be fine.

Input Levels

When using random or pseudorandom noise signals, -12 to -15 dB or so is the preferred input level for any kind of measurement in Smaart including IR measurements. -12 dB is the point where the input levels in Smaart turn yellow.

Delay Time

When using period-matched noise as your excitation signal you can set the delay time to 0 if you don't already have it set for your selected signal pair. If you do, then there's no harm in leaving it alone. If you are using a random noise source and don't already know the delay time for your signal pair, run the IR measurement once to find it, then click the "Ⓟ" button to set it, then run the measurement again.

Appendix A: Applicable Standards and Further Reading

Further Reading on Audio Engineering and Acoustics

Ballou, Glenn. ed. *Handbook for Sound Engineers.* Focal Press.

Beranek, Leo J, Music, Acoustics and Architecture. Wiley.

Berg, Richard E. & Stork, David G. *The Physics of* Sound*.* Prentice Hall.

Borden, G. J. & Harris, K. S. Speech Science Primer: Physiology, Acoustics and Perception of Speech. Williams and Wilkins.

Davis, D., Patronis, E., & Brown, P. *Sound System Engineering 4e.* Taylor and Francis.

Jones, D.S. Acoustics and Electromagnetic Waves. Clarendon Press.

Kleppe, J.A. Engineering Applications of Acoustics. Artech House.

McCarthy, Bob. Sound Systems: Design and Optimization: Modern Techniques and Tools for Sound System Design and Alignment. Focal Press.

Olson, Harry F. *Modern Sound Reproduction.* Van Nostrand Reinhold.

Rigden, John. Physics and the Sound of Music. Wiley.

Rossi, Mario. Acoustics and Electro-acoustics. Artech House.

Toole, Floyd. Sound Reproduction, Loudspeakers and Rooms. Focal Press.

Applicable Standards for IR Measurements and Speech Intelligibility

Several of the techniques, procedures and quantities discussed in the second half of this document are the subject of ISO and IEC standards. We highly recommend actually reading those standards, rather than relying on our interpretations and synopses.

ISO 3382, Acoustics — Measurement of room acoustic parameters, parts 1 and 2, provide a lot of additional information and recommended practices in for measurement and analysis of reverberation time and other objective metrics commonly used for evaluating room acoustics.

- Part 1: Performance Spaces, (ISO 3382-1) is geared more toward concert halls, opera houses and theaters, and contains a quite a bit of information on acoustical quantities other than reverberation times not found in…
- Part 2: Reverberation time in ordinary rooms (ISO 3382-2)

We note that there is quite a bit of overlap between the two. In fact, the rationale for making it a two-part standard doesn't seem immediately obvious to us, but if you didn't want to buy both parts, part one is the more comprehensive of the two. Both parts contain much more information with regard to measurement procedures and statement of results than we have presented here.

For anyone interest in getting into the nuts and bolts of techniques used for indirect IR measurement, ISO 18233, Acoustics — Application of new measurement methods in building and room acoustics is a good place to start. If you are interested in making speech intelligibility measurements using STI or STIPA, IEC 60268-16, Sound system equipment – Part 16: Objective rating of speech intelligibility by speech transmission index (Edition 4, 2011 or later) is really a must-read.

Appendix B: Room Volume, Absorption and Reverberation time (RT60)

Reverberation time is usually measured or calculated for a 60dB decay range (RT60). The following information will help you to appreciate the factors that affect reverberation time so that, after measurements have been made, appropriate changes can be suggested where necessary.

RT60 is heavily dependent on the room volume, the amount of absorption in the room, and air absorption at higher frequencies. A popular approximation, based on Sabine's formula, is:

$$RT = {}^{55.3V}\!/_{c(A + 4mV)}$$

where

- RT is the time it takes for reverberant sound energy to decay by 60dB (in seconds)
- V is the volume of the room in cubic meters
- c is the speed of sound in meters/second (varies with temperature)
- A is the total sound absorption of room materials in square meter Sabines – i.e. the sum of surface areas, each multiplied by its respective sound absorption coefficient
- m is the intensity attenuation coefficient of air per meter (varies with temperature and humidity)

Typical Material Absorption Coefficients

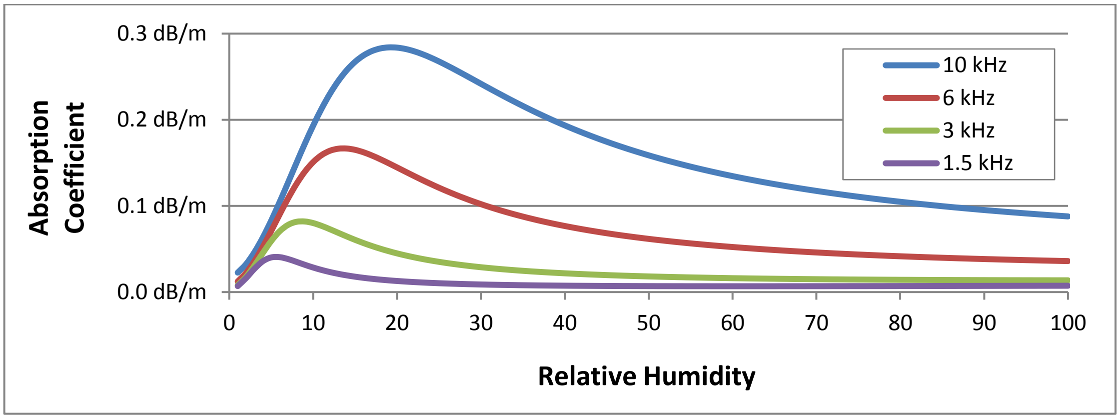
Material Type	**Typical mid-band sound absorption coefficient**
Marble	0.01
Plastered walls	0.02
Bare brick	0.03
10mm Plywood	0.09
10mm mineral wool	0.60
25mm polyurethane foam	0.70
Acoustic ceiling tile	0.72
Audience member on upholstered seat	0.88

Note that absorption coefficients vary with frequency.
See: www.sengpielaudio.com/calculator-RT60Coeff.htm

Air Absorption

Air absorbs sound quite significantly at higher frequencies due to molecular resonance. This high frequency air absorption varies with temperature and relative humidity in a way too complex to cover here. However, a useful air absorption calculator may be found at the link below.

http://resource.npl.co.uk/acoustics/techguides/absorption/



Air absorption (in dB/m) vs relative humidity at room temperature

Note that sound absorption through air is per meter – not per doubling of distance as seen for radial attenuation. Sound in a room with a mid-band RT60 of, say, 2 seconds, will travel several hundred meters before decaying into the noise floor so air absorption is a significant additional factor in reducing high frequency reverberation times.

Schroeder Cut-off Frequency

RT_{60} assessments are usually restricted to the Schroeder region of the frequency range where mid and high frequency wavelengths are short compared to the room dimensions, and sound paths become dense, diffuse and fairly random. This is often referred to as the statistical or stochastic region. At lower frequencies individual modes occur. These modes are more related to room dimensions, less dense and don't follow the diffuse reverberation characteristics required for accurate reverberation time analysis.

The crossover point between the stochastic and modal regions, which Manfred Schroeder called critical frequency and everyone else now calls the Schroeder Frequency, marks a cross-over region where modes start to overlap by a fairly arbitrary factor of three – it is not an exact science. The popular approximation published by Schroeder in 1962 is:

$$f_c = 2000\sqrt{RT/_V}$$

As before, *RT* is the time taken for the reverberant sound energy to decay by 60dB (in seconds) and *V* is the volume of the room in cubic meters. For room volume in cubic feet, the equation can be written as:

$$f_c = 11885\sqrt{RT/_V}$$

Practical Measurement Considerations

Most of the above equations are approximations assuming an omnidirectional sound source, homogenous sound distribution – and the possession of accurate venue information! In reality, accurate venue information may be difficult to acquire and operating conditions may vary from event to event. Reverberation characteristics are also likely to vary slightly across the audience area and early decay characteristics (see later) will vary significantly from seat to seat so accurate measurements, over a practical range of operating conditions, are recommended.

As most rooms have significant background noise levels (the noise floor), a full 60dB decay range isn't always available – especially when the direct sound is at normal speech levels or, as is possible with Smaart, we measure a system with an audience in place.

For convenience, instead of measuring the full 60 dB decay time, the decay slope (in dB per second) is measured over a 30 dB range (preferred) and then doubled to work out the RT60. The slope is conventionally measured between the -5dB and -35dB points, as long as the lower point is at least 10dB above the noise floor. If the noise floor is exceptionally high, it may sometimes be necessary to measure the decay slope over a smaller level range. In that case a 20 dB range may be measured instead and then tripled to normalize to equivalent 60 dB decay time.

According to ISO 3382, reverberation time measured over a 30 dB range is called T30 and T20 signifies a 20 dB measured range. In their notation scheme, the letter T, by itself, stands for 60 dB decay time and so hopefully it is understood that the 20 and 30 refer only to the measured range. The stated reverberation time in seconds for either figure is the equivalent 60 dB decay at the measured rated of decay.

Typical values are as follows:

Talk studio	less than 0.5 seconds
Cinema/auditorium	0.4 to 1 seconds
Conference/classroom	up to 1 second
Musical theatres	1 to 1.5 seconds
Chamber music/opera venues	1.5 to 2 seconds
Symphony halls	1.5 to 2.5 seconds

(In practice, of course, acceptable ranges vary with expectations – based on the venue size)

Appendix C: Sound Source Characteristics

A sound source's positioning, orientation and directional characteristics will determine both its ability to cover an audience consistently and its propensity to excite non-audience areas of the room. The latter can enhance the listening experience if, for example, it adds a nice, clean "tail" of diffuse reverberation to an orchestral performance. But room excitation can also ruin the auditory experience if, for example; loudspeaker lobes are causing coloration or if loudspeaker arrays are poorly aimed causing late reflections, excessive reverberation and a loss of intelligibility.

Ignoring high frequency air absorption, a point source's direct sound pressure will be inversely proportional to the source-to-listener distance (i.e. 1/r or 6dB attenuation for every doubling of that distance). In echoic rooms, however, reverberation will add to the direct sound at the listener position depending on how directional the source is. An omnidirectional source, away from any boundaries, would radiate sound pressure spherically and cause strong room excitation.

Directivity Factor (Q)

If a sound source is directional, so that its coverage is not fully spherical, its sound power gets concentrated into a smaller sector of a sphere. To help quantify that sound power concentration, sound sources are said to have a directivity factor (Q).

Directivity factor (Q) is:

- 1 for spherical source
- 2 for hemispherical source
- 4 for quarter spherical source…and so on.

Sound Pressure Level at the Listening Position

If we use directivity factor (Q) to quantify a sound source's directional properties, then, for on-axis listeners, the directivity-dependent relationship between source sound power level and the total (direct + reverberation) listener position sound pressure level approximates to:

$$L_p = L_w + \log_{10}\left({}^{Q}/_{4\pi r^2}\right) + ({}^{4}/_{R})$$

The directivity factor (Q) can also be used to estimate the critical distance (Dc).

$$Dc = 0.14 \times \sqrt{(QR)}$$

For the previous two equations:

L_p is the total (direct + reverberation) sound pressure level at the listening position
L_w is the source's sound power level in dB referenced to 10^{-12} Watts
Q is the directivity factor (1 for spherical, 2 for hemispherical, 4 for quarter spherical etc.)
π is 3.142…
r is the distance between the source and the listening position
R is the room constant (the room's ability to absorb sound – i.e. the product of surface area and absorption coefficient)

We don't need to memorize these equations but understanding the relationship will help us trouble-shoot problems.

For instance, we see that:

- Adding absorption will, obviously, reduce reverberation and increase the critical distance. Alternatively, we could reduce the effects of the reverberation by aiming highly directional loudspeakers into the audience but away from the walls and ceiling
- Conversely, using less directional sources and/or lower room absorption (preferably with plenty of diffusion and a good balance between early and late decay times) will add "air" to our sound and prevent it from becoming too "dry."

Directivity index (DI)

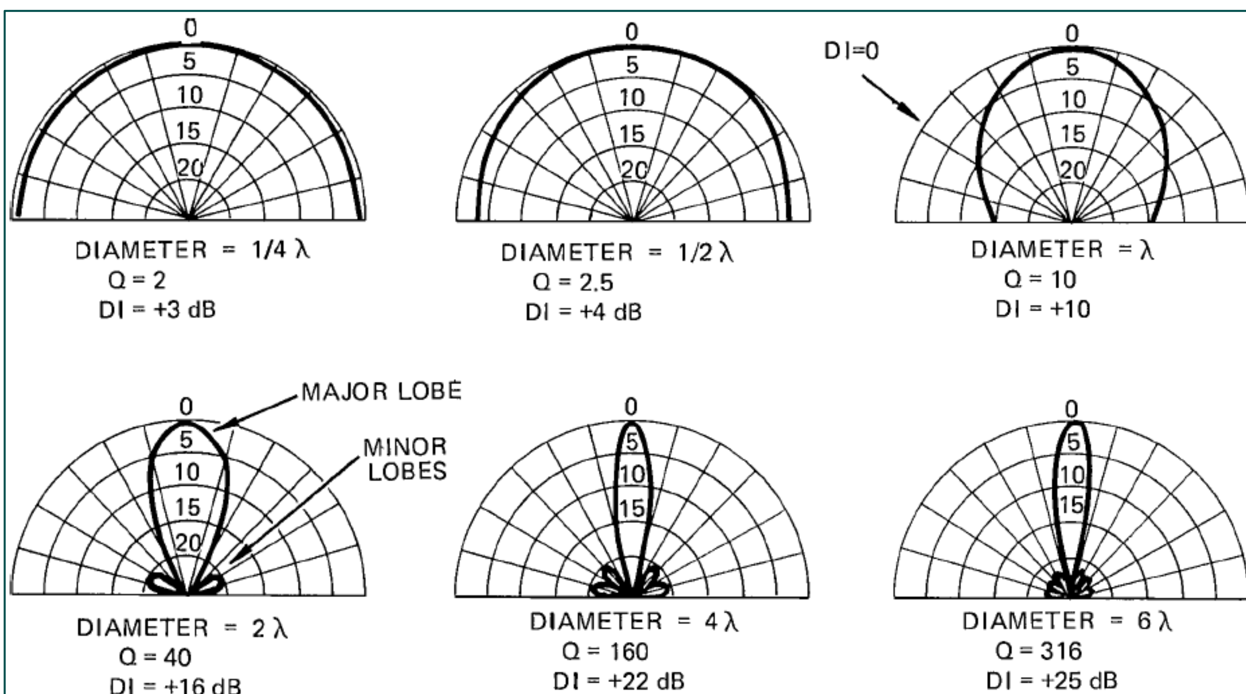
An alternative directivity figure that indicates the concentration of radiated power in dB is the Directivity Index (DI). Its value (in dB) is:

$$DI = 10 \times logQ$$

So, assuming the listener is on axis, if Q = 2, DI = +3dB; if Q = 4, DI = +6dB etc…

Q and DI figures in Practice

Large radiating surfaces tend to be more directional than small ones. Good far-field summation occurs perpendicular (on-axis) to large sources, less than perfect summation occurs off-axis due to phase variations between waves emanating from different areas of the surface. This can cause minor off-axis lobes at some frequencies and cancellations at others.



Courtesy: JBL Professional/Harman

The above illustration shows the polar patterns, plus the relevant directivity factor (Q) and directivity index (DI) of baffle-mounted piston-like sound sources (e.g. stiff loudspeaker cones) for various diameters with respect to the wavelength of sound being produced.

Reading from left to right, top to bottom, you can either think in terms of an increasing diameter at a fixed frequency for each successive example . . . or, you can think in terms of increasing frequency for a fixed diameter for each successive example.

The nice thing about the directivity index figure DI is that it can be used in conjunction with a source's polar plot (assuming a dB level scale) to indicate the true directivity index for an off-axis listener. This is shown in the above top right example. The on-axis DI is +10dB, but, at the off-axis point (arrowed) the polar response is 10dB down with respect to the on-axis position. The DI at the off-axis point arrowed is the on-axis DI minus the off-axis figure – i.e., the off-axis DI is +10dB - 10dB = 0dB.

Conventional Loudspeaker Arrays

A spherical array of identical high-Q loudspeaker horns will act like one large spherical radiator if the optimum inter-cabinet splay angles are used. Note, however, that the overall array directivity will tend to be lower than that of the individual horns at mid and high frequencies due to the array's wider overall coverage.

At lower frequencies, however, arraying can provide tighter pattern control by making the system acoustically larger. Things can get quite complicated and pretty difficult to predict – especially when you factor in manufacturing tolerances, marketing-optimized loudspeaker specifications, inaccurate venue drawings and a lively room's sensitivity to relatively small changes in array shape and tilt.

Prediction software can get us close to our design goal and will certainly enable us to work out requirements and budgets etc., but we need Smaart to check that installed systems are meeting spec. Smaart's ability to complete as-built measurements – even during shows with an audience in place, if required – makes it a very powerful verification tool.

"Line" Arrays

It should be noted that the term "line" array has expanded in popular usage in recent years to become a term for a relatively long, vertical array of loudspeaker – typically just one cabinet wide. Manufacturers often imply that a line array system will give the user a text-book line source radiation characteristic and many users assume they'll benefit from a radial attenuation of 3dB per doubling of distance rather than the 6dB/doubling of a conventional array.

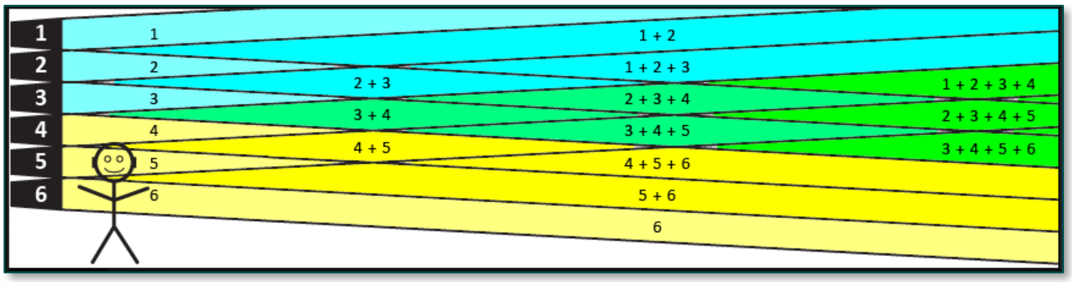
In reality, there is rarely the budget for a straight line source column as it would have to stretch from the floor to the highest seat. Most "line" arrays are much shorter for budget and safety reasons so have to be progressively curved from top to bottom in order to cover the seating areas from the furthest, highest seat down to the front floor areas beyond the front fill coverage.

However, the upper, straighter part vertical pattern control can be very tightly controlled at mid and high frequencies. And, if aimed hard into the audience, can reduce upper walls and ceiling excitation, reducing the reverberant level and increasing the critical distance.

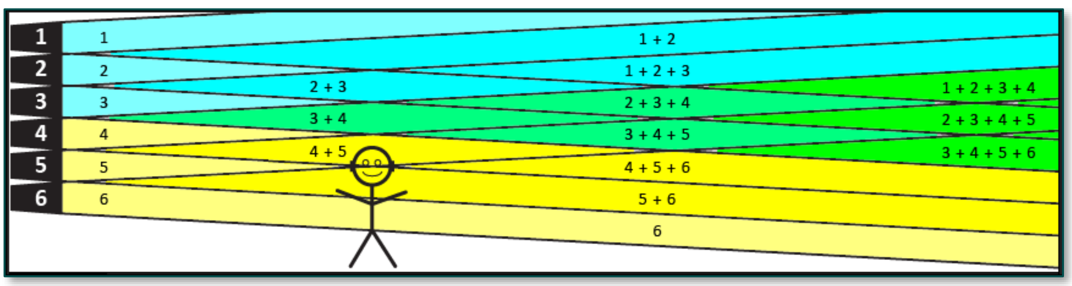
A Note about Echoes

The upper part of a line array provides tight mid-high pattern control above the array – minimizing direct ceiling excitation. However the low radial attenuation rate can make them prone to create very audible echoes from distant boundaries if they aren't aimed accurately.

Always use Smaart's linear (Lin) IR graph to check for echoes as they can often be hidden amongst the room's diffuse reverberation – only to become audible, especially on stage, when the audience is in place and reverberant energy levels have dropped.



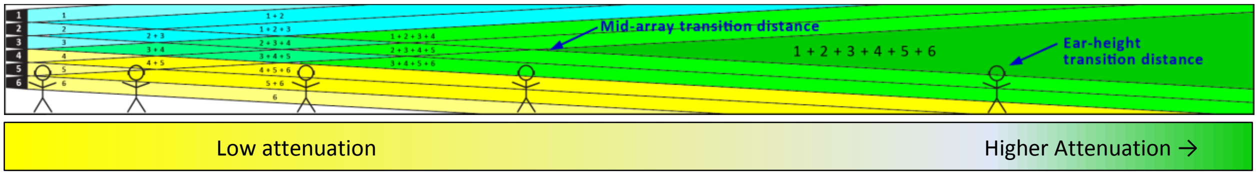
A simple way of understanding the line array effect of reducing radial attenuation from 6dB/doubling of distance to, perhaps, 3 or 4dB/doubling of distance at mid-high frequencies is to imagine a listener very close to just one horn-loaded element of the array. In the figure above, the listener (let's call him Victor) mainly hears element 5, with some minor off-axis contributions from the other elements.



Each element (#1-6) is a point source and would, if measured in isolation, have the usual 6dB/doubling of distance radial attenuation characteristic. However, as Victor moves further away from element 5, he starts to benefit from the vector sum of elements 4 and 5 so the usual 6 dB/doubling characteristic is modified by the extra element coming into play. This phenomenon is similar to the near field effect, mentioned earlier, which occurs close to any large sound source.

As Victor continues his journey further away he then benefits from the vector sum of more and more elements and these partially compensate for the usual 6 dB/doubling of distance each element in isolation would have. Victor will benefit from this low (3-4 dB/doubling of distance) attenuation region all the time there are extra elements available.

However, Victor will eventually run out of these extra elements and the radial attenuation characteristic will revert back to the usual 6dB/doubling of distance. This point in Victor's journey is called the *transition distance* – see next illustration.



Note that the transition distance, for a true line source, is proportional to frequency and to the line length[2].

$$Transition\ Distance = \frac{Line\ length^2\ x\ frequency}{2\ x\ speed\ of\ sound}$$

Allowing for the fact that the speed of sound is temperature-dependent, transition distances for a typical 20 ft (~6m) straight long line source at 68° F (20° C) will be approximately:

Frequency	200Hz	400Hz	800Hz	1.6kHz	3.6kHz	7.2kHz
Transition distance (ft)	35.5	71.0	142	284	638.6	1277
Transition distance (m)	10.8	21.5	43.3	86.5	194.7	389.3

This suggests that high frequency propagation would be very efficient. But, in practice, HF energy will be significantly reduced by air absorption (by up to 0.1dB/meter, worse case) and, of course, imperfect coupling at high frequencies due to rigging tolerances etc.

It is particularly important to remember that transition distance is proportional to the square of the line length if spectral balance is to be maintained over a wide area. The length of the straight line section (at the top of most arrays) needs to be at least 6% of the maximum distance to be covered for spectrally balanced vocals.

Note that the theoretical mid-hi coverage pattern for a straight-line array narrows to a point (see figure above) at the *mid-array transition distance* (remember that this varies with frequency), hence our need to curve real-world arrays for practical audience coverage. This is especially true where some audience members (like Victor) are below the maximum mid-array summation point – Victor's *ear-height transition distance* is well beyond the mid-array transition distance in this case.

In practice, any temperature and wind gradients will move these transition points around significantly.

Beyond the transition distance there are no more elements available to compensate for any increase in listener distance so the line array effect ceases to operate and the coverage "defocusses" towards point-source like behavior, with its characteristic 6dB-per-doubling radial attenuation characteristic.

Q: What has all this got to do with measuring room acoustics using Smaart's Impulse mode?

A: Smaart's ability to measure room phenomena "live" with an audience in place means that you will often be asked to assess acoustics under practical, amplified show conditions using an installed PA system. Many of these installed systems will be "line" arrays – whether appropriate or not – and a basic understanding of them will allow you to make intelligent recommendations.

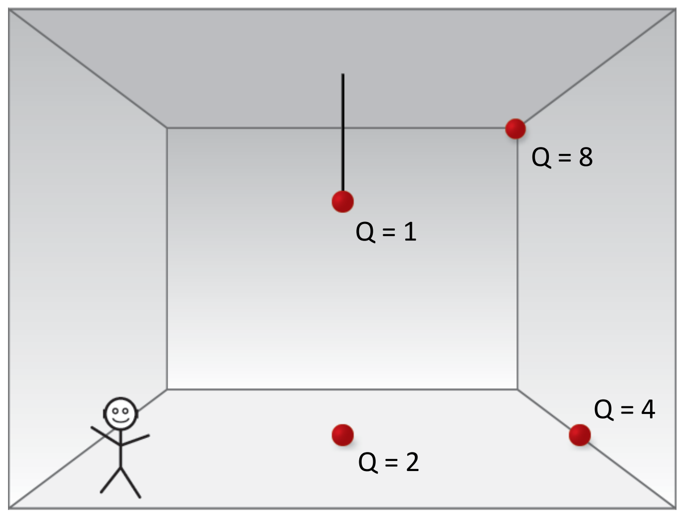
Appendix D: Boundary effects

Acoustically Small Sources

Acoustically small sound sources are sources whose dimensions are small compared with the wavelengths they produce. Away from boundaries, acoustically small sources tend to be omnidirectional.

A single-driver 1m cube subwoofer would be a good example of an acoustically small sound source as its size would be less than a quarter of the wavelength at low bass frequencies. At 80Hz, for instance, the wavelength would be just over 14 ft (~4m).

The effective directivity of an acoustically small source tends to be governed by local boundaries. The illustration shows a large room with some identical, acoustically small (red spherical) sources in various positions with respect to the room boundaries. (Ignore Victor. He's only there to avoid 3-D ambiguity.)



- The source that is dangling in free space has a Directivity Factor (Q) of 1 and radiates its acoustic power spherically – into full space.
- The mid-floor source (Q=2) has its acoustic power concentrated into a hemisphere. It radiates the same power but concentrated into half-space.
- The baseboard source (Q=4) has its acoustic power concentrated into a quarter-sphere. Again, it radiates the same power but, this time, concentrated, into quarter space.
- And the rear corner source (Q=8) has its acoustic power concentrated into an eighth-sphere – radiating the same power but, this time, concentrated, into eighth space.

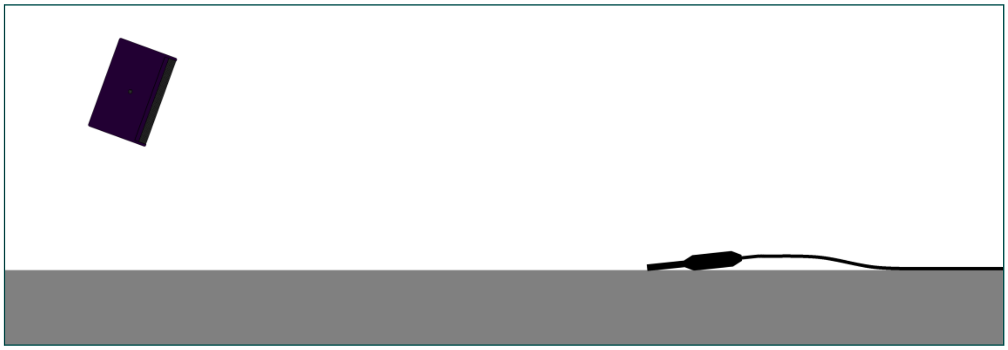
Useful on-axis free field sound pressure level/headroom increases, with respect to Q=1, will approach 6dB per boundary:

- +6dB for Q=2 (half space)
- +12dB for Q=4 (quarter space)
- +18dB for Q=8 (eighth space)

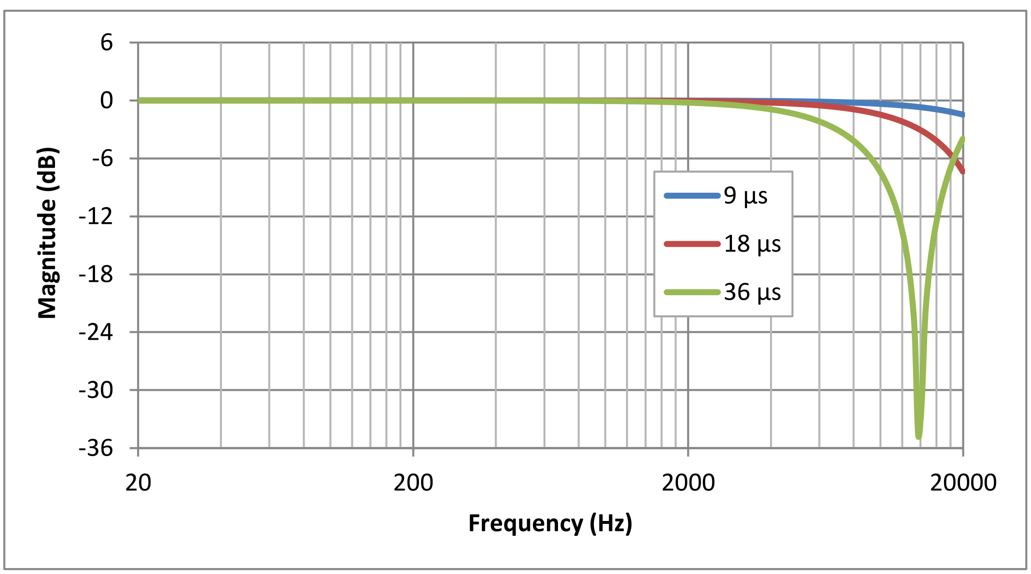
Microphones Near Boundaries

Similar SPL increases apply to pressure microphones if they are placed acoustically close to a surface – i.e., much less than 1/6th of the shortest wavelength of interest. At 1/6th wavelength, the "round trip" will be 1/3rd wavelength, causing a 120° phase shift and resulting in unity gain summation. Larger spacing would cause partial or full cancellation and combing.

Placing a microphone capsule on the center of a large surface can be useful when measuring, for instance, PA systems in an empty room devoid of seats, as the reflective qualities of the floor will only be visible as a 6dB increase in overall level.



Placement of the microphone in relationship to the floor is critical in this type of measurement, as the reflection from the floor still causes a comb filter. The object of the exercise is make the path of the first reflection so short, relative to the path of direct sound, as to push first null in the resulting comb filter well above the audible spectrum. If the path of the floor bounce is more than a few millimeters longer than the path of direct sound, the fist null of resulting comb filter will be low enough to produce at least a visible lowpass filter function in the top octave, if not an actual null within the audible spectrum.



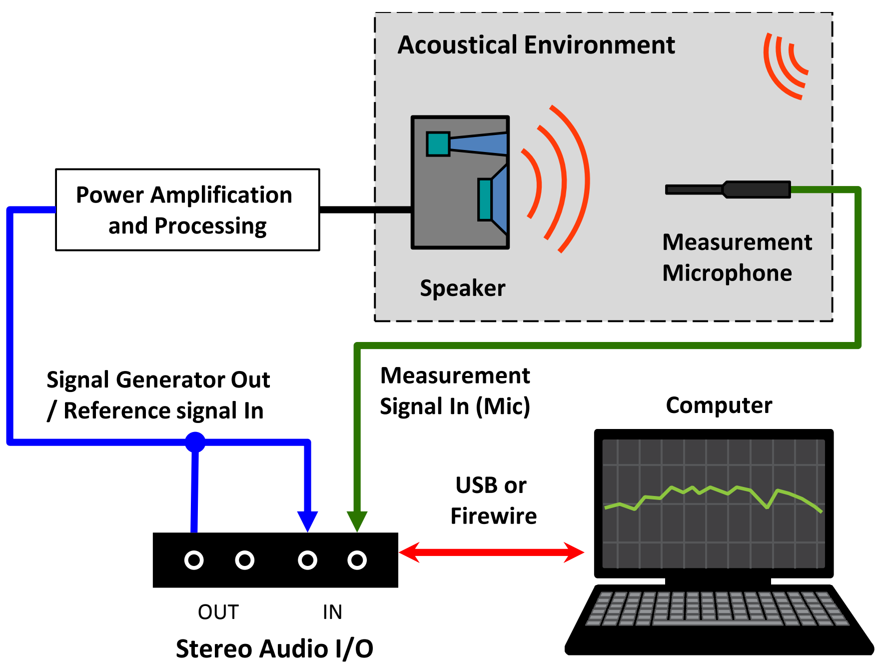
The chart above shows comb filter functions for 9, 18 and 36 µs reflections respectively, which would equate to reflected path lengths approximately one eighth, one quarter and one half inch (3, 6, and 12 mm) longer than the path of direct sound. A common strategy for minimizing this issue is to rest the barrel of the microphone on a coil of cable so that the capsule is actually in contact with the floor or very nearly so. It may also be preferable to use a small diaphragm microphone to further limit the maximum possible length of the floor bounce path.

Appendix E: Typical Measurement Rig Set-Up

The following are some example measurement-system setup diagrams for transfer function and dual-channel IR measurement in Smaart. Dual-channel measurements are made by comparing a *reference signal* (system input) and a *measurement signal* (system output). They are an essential tool for aligning loudspeaker systems because unlike “time-blind” single-channel RTA measurements, dual-channel measurements can show you both the magnitude and phase response of a system – that is, both energy and timing. Additionally, the same pair of signals (reference and measurement) can be used to calculate the coherence function, an assessment of the linearity of a system that can provide important clues about signal-to-noise ratio, reverberance and overall quality of your measurement data.

Stereo (2x2) Audio I-O

In this example, Smaart’s internal signal generator is used to excite the system under test. The signal generator is assigned to output 1 of a 2-in/2-out audio input-output (I-O) device, labeled *Stereo Audio I-O* in the diagram below. The audio I-O device is connected to the computer via USB or Firewire. Output 1 on the audio I-O is connected to the input of the system under test and also routed back to input 1 on the audio I-O using a Y-split cable (a hard-wired loop-back). Input 1 on the audio I-O will be assigned as the reference signal of a transfer function measurement in Smaart. A measurement microphone connected to input 2 of the I-O device provides the measurement signal.



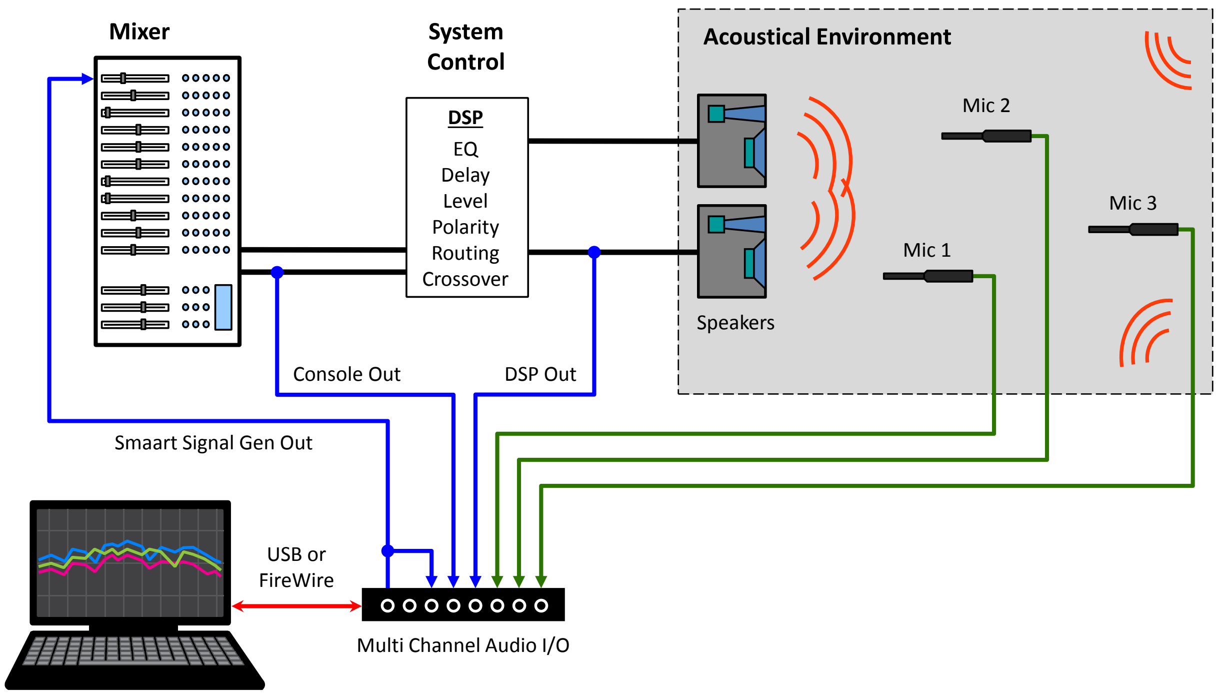
Basic connections and routing should be pretty much the same for any audio I-O device you may encounter. We trust that you understand the audio cabling and connections necessary connect your equipment together. Note that the audio I-O device in this case could conceivably be the computer’s built-in stereo line input and headphone output, in conjunction with a self-powered measurement microphone or external mic preamp and phantom power supply and a little bit of creative cabling.

Multi-Channel I-O

In version 7, Smaart introduced the capability to run and display as many simultaneous real-time spectrum and transfer function measurements as your computer can handle. Having a multi-channel I-O device enables you to set up multiple microphones to compare different measurement positions in real time, without a lot of running around. This can be a huge advantage when working with larger, more complex sound systems.

Traditional USB/FireWire Stand-Alone Interfaces

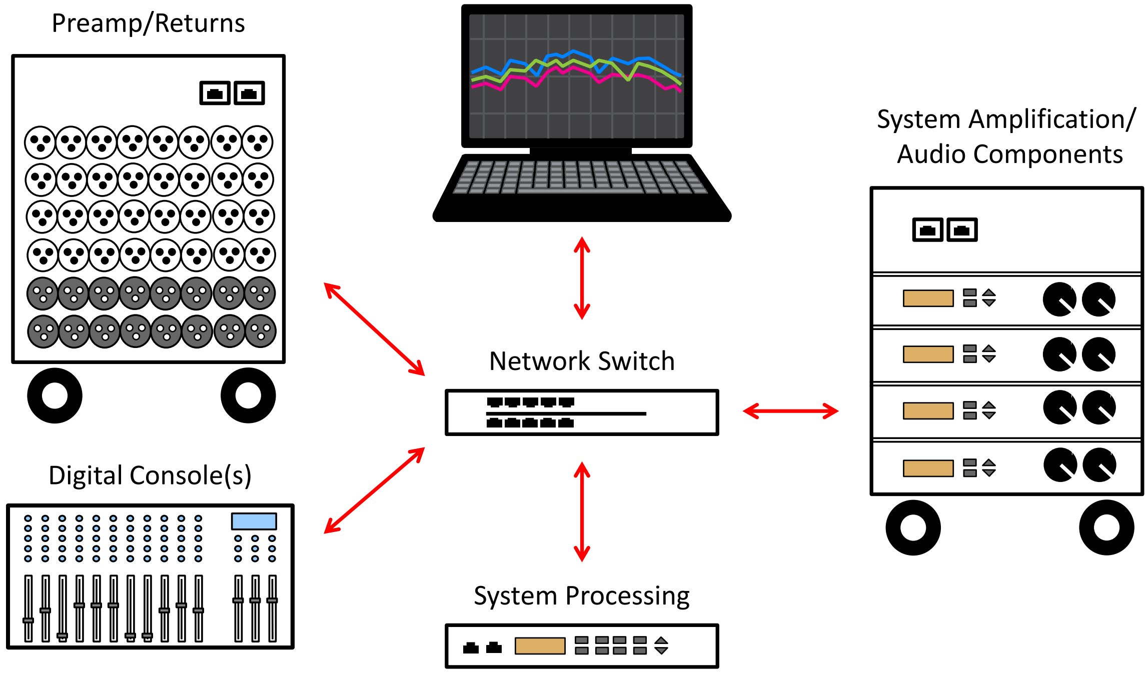
Eight-channel, single rack space packages are a common form factor for multi-channel audio interfaces. The single rack space format makes portability in a rack case, or in your backpack, equally viable. In addition to fielding more microphones, having more audio inputs to the computer enables you to tap into multiple electronic measurement points in a system.



In the diagram above, we have three measurement microphones set up, along with electronic measurement points pre-system-processing and pre-loudspeaker-system. There is also a hard-wired loopback pre-console. The *Console Out* and *DSP Out* can be used as reference signals that let you isolate the response of the system post-console or post-DSP. Alternatively, they can be used as measurement signals for analyzing the console, EQ, or crossover response in Smaart.

Network Audio I-O

The ability to digitize an audio system from end-to-end, from stage pre-amps, all the way to the loudspeaker system(s) has become a reality in the past few years. In some professional sound systems, the only analog signals present are those from the microphones to the stage conversion box, where they are digitized and then distributed wherever they are needed via the network.



In this scenario, all you might need in terms audio I/O hardware for Smaart is a way to connect your computer to the network. Using the Audinate® Dante® network protocol as an example, a single connection to your computer's Ethernet port makes any signal from any source connected anywhere on the network potentially available for analysis in Smaart and enables you to send test signals from Smaart to any destination on the network that is capable of receiving an input stream. The Dante Virtual Soundcard software application presents a selected group of audio streams from the network to Smaart as a standard, multi-channel ASIO or CoreAudio device. Smaart sees the virtual I-O device, just as it would any locally connected USB or Firewire audio I-O. Selected audio streams from the network appear as input channels on the virtual I-O device and Smaart's signal generator can be assigned to its output channels. Routing is done within the Dante Controller application.

Audinate and Dante are trademarks of Audinate Pty Ltd.

Appendix F: Licensing and Installation

The licensing system deployed with Smaart v.8 provides you with greater control of your activated systems than any previous Smaart version. The information in this section can help you master the license management tools in Smaart and on our web site so you can spend more time measuring sound and less time messing with licensing. If you can't find the answers you are looking for here, feel free to send an email to support@RationalAcoustics.com, or give us a call at +1-860-928-7828 Mon-Fri, 9 AM-5 PM EST(UTC -5).

my.RationalAcoustics.com

If After purchasing Smaart v8, you will receive an email (or CD-ROM) containing your Smaart license code and the program installers. Before installing Smaart for the first time, you must create an account at http://my.RationalAcoustics.com/ (if you don't already have one).

Once you have an account, you can register your license by clicking the "Register a new Smaart license" button on the Account Details page, or by installing and activating Smaart (via "Activate Online") on a computer using your new license number and your account login.

Installing Smaart v8

The Smaart installers work like any other installer for Windows and Mac operating systems. The same installers can be used on both 32-bit and 64-bit operating systems.

Software Installation on Windows®

Initial installation of Smaart v8 on the Windows operating system is done using a setup program that operates very much like virtually any other software installation program for Microsoft Windows. Note that on Windows 7 and newer systems, administrator authority is required to perform the installation. Other than that, you only need to read and agree to the End User License Agreement (EULA), confirm selection of the folder where the program will be installed, and choose whether or not to have the installer program create a shortcut for Smaart on your desktop.

Software Installation on Mac OS X®

Smaart v8 for Mac OS X is supplied in the form of a Mac application bundle, so installation is simply a matter of dragging the Smaart v8 icon into your Applications folder. Note that Smaart v8 is distributed in a disk image file that requires you to agree to the End User License Agreement (EULA) before you can access to the software packed inside. Once you have read and agreed to the EULA you can drag the Smaart v8 icon into your applications folder.

Activating an Installation

When you run Smaart for the first time, an activation screen will appear. You need to activate your installation before you can use the software.

There are four basic requirements for activating a Smaart v8 installation:

- A valid Smaart v8 license code (XXXXXX-XXXXXX-XXXXXX).

- A license management account at my.RationalAcoustics.com
- One or more installation spots available on your license.
- Internet access on or near the computer you are trying to activate.

Online Activation

If Smaart detects an internet connection, the Online Activation prompt will appear and you can activate without leaving the program. You'll need your 18-digit alpha-numeric Smaart v8 license code and your my.RationalAcoustics.com login information to complete the online activation.

Off-line Activation

If you need to activate Smaart on a computer that is not connected to the Internet, you can manually register the Smaart machine ID from within your account at my.RationalAcoustics.com.

Open Smaart on the computer that is not connected to the internet and the Smaart machine ID will be displayed on the first screen that comes up. Clicking the *Machine ID* will copy it to your clipboard.

From any computer/device that is connected the Internet, open a web browser, navigate to http://my.RationalAcoustics.com, and log in to your license management account.

Once logged in, click the **Your Software Licenses** link at the top of the page to view your registered Smaart licenses. If your license isn't listed here, it may not be registered. In that case, click **Register a new Smaart license** towards the top of the page to proceed.

Click on your v8 license code and you are brought to a page where you can see the total number of installations allowed on your license, the number of installations you have used, and how many are still available.

Assuming that you have a least one installation spot available, click the **Offline Activation** button. Next, enter your machine ID, the name and e-mail address you want to associate with this installation, and a friendly name to identify the computer. There is also a field for a Block Code, which you can ignore unless you are reactivating an installation on a computer that was previously deactivated (more on this below).

When you finish entering the required information, click the **Submit** button to get your Activation Code. Go back to Smaart on your offline machine, enter the Activation Code, and click the **Activate** button. If the code is correct, you will see a success message.

A note about company-owned licenses

For organizations with multi-user licenses, Offline Activation allows users to perform their own software installations without exposing the credentials required to administer the license.

Once the user installs Smaart, they can click "Offline Activation" and send their machine ID to the account administrator, who can then register the machine ID to the Smaart license via my.RationalAcoustics.com and send the provided Activation Code back to the user.

Deactivation

Smaart v8's licensing system is equipped with a "Deactivate" feature that allows you to reclaim an installation spot if a computer is being retired, temporarily replaced, or reformatted. If your file system is being migrated or backed up to a drive image, there are some important considerations to make with regards to Smaart's licensing, please see the section on *Time Machine, Migration Utilities, and Cloning Software* before deactivating. If you are upgrading your operating system, please read the section titled *Upgrading your Operating System or Major System Components.*

Moving Smaart to a New Computer -*or*- Clean Reinstall of your Operating System

If you need to move Smaart from one machine to another, or if you are retiring an old computer, you will need to Deactivate (or "Block") your current installation. Deactivating an installation renders it unusable on that machine until/unless it is reactivated.

If you are completely reformatting your computer, you must deactivate Smaart first to regain your installation spot. If you plan to use **Time Machine**, migration utilities, or any kind of cloning software to move your file system to a new machine (or new hard drive), please see the appropriate section below.

To deactivate a Smaart v8 installation go to the "About" window for the program and click the **Deactivate Installation** button.

After clicking the deactivation button, and confirming that you really want to deactivate, Smaart will attempt to contact our server and complete the deactivation. If the attempt was successful, then you don't need to do anything else. If Smaart can't contact our server, you will see a screen similar to the one below.

If Smaart was unable to connect to our web server, make a note of the Block Code and machine ID then open a Web browser and go to my.RationalAcoustics.com to complete the deactivation.

The process is as follows:

1. Login to your account at my.RationalAcoustics.com and click the link for *"Your Software Licenses"* on the navigation bar at the top of the page.
2. Click on your Smaart v8 license number and find the machine ID that you wish to deactivate in your list of current installations.
3. Click **Release** in the Actions column.
4. Enter the Block Code from Smaart along with your name and e-mail address in the fields provided and click the **Submit** button. Your available installations will increase by 1 following a successful deactivation.

Note that your Machine ID and Block Code are also displayed on the activation screen that appears if you attempt to run Smaart again after deactivation. If you inadvertently closed the screen show above without recording those numbers or you write down one of them incorrectly, you can always view them again. We recommend that you do not uninstall Smaart on the deactivated machine until you have confirmed the deactivation.

Time Machine, Migration Utilities, or Cloning Software

If you plan to restore a backup environment – OS, software, etc. – to a new or reformatted hard disk, or migrate your files and software to a new computer using a system migration tool, it is extremely important that you make your backup image or copy your files to the new computer or disk drive **before** deactivating Smaart. Once you've made the backup image, or migrated files to the new hard drive/computer, go back and deactivate on the old system. This ensures that the deactivation is not transferred to the new hard drive or new computer. If you start Smaart on the new hard drive/computer and encounter an "Error 523", please see the section below titled *Reactivating after restoring from a backup or migrating system files.*

Upgrading your Operating System or Major System Components

If you plan to upgrade your operating system (**Win 7** → **Win 10**, **10.8 → 10.10**, etc.), but you are keeping the file system intact (not reformatting), *-or-* if you are replacing major system components such as the RAM, video card, or motherboard, it is better to leave Smaart activated and email us if you run into any problems (support@RationalAcoustics.com).

Reactivation

Reactivating Smaart on a Deactivated Computer

If you attempt to run Smaart on a machine that has been deactivated, you will see a screen very much like the original activation screen shown earlier in this document, but with the addition of a **Block Code**.

If you have not made any hard drive or operating system changes to the computer since deactivation, the process for reactivating Smaart is identical to the initial activation process if you choose to use the "Activate Online" option. If you are reactivating a computer that is not connected to the internet, you will follow the steps above for activating offline, but in addition to your machine ID, you will also need to enter the Block Code listed on the first activation screen. Once you have submitted the necessary information, you will receive a new Activation Code to reactivate Smaart.

Reactivating after restoring from a backup or migrating system files

If you are activating an installation that's been restored from a backup or migrated from another machine the process is essentially the same as a new installation.

If you receive **Error 523** while trying to activate, you need to delete the old Ticket file (licensing file) to get back to the initial activation screen in Smaart. The location of the Smaart v8 ticket file depends on the operating system version, please look for your operating system in the list below to learn the appropriate file path. After Ticket deletion, start Smaart. If you are asked to restore a missing file, click "No". Once the Activation window appears, try activating again.

Mac OSX: (harddrive)\Users\Shared\Ticket\Smaart8.ticket

Windows Vista\7\8\10: C:\Users\Public\Ticket\Smaart8.ticket

If you encounter any licensing problems, please send an email to support@rationalacoustics.com, or give us a call at +1-860-928-7828 Mon-Fri, 9 AM-5 PM EST (UTC -5).

Glossary of Installation and Activation Terms

License Code	The 20-digit alphanumeric number that identifies your Smaart v.8 license. This license is registered to your account at my.rationalrcoustics.com, allowing you to download installers and activate installations.
Machine ID	The unique number assigned to your computer configuration by Smaart. If Smaart is not yet activated, the Machine ID for your computer can be found on the first activation wizard screen that appears when you run Smaart for the first time after installation. If Smaart is activated, the machine ID can be found in the "About" menu. Each unique machine ID can be activated/deactivated up to 9 times before reaching its limit.
Activation Code	The code that activates Smaart, either initially, or after a previous Deactivation. This code can be obtained by manually registering your machine ID through the web interface at my.rationalacoustics.com. If you have already activated Smaart, the Activation Code can be found by clicking on the appropriate machine ID on your v.8 license page at my.rationalrcoustics.com.
Block Code	The code obtained if Smaart cannot communicate a deactivation attempt to our web server. The Block Code can be used to manually "Release" a Machine ID from your license. If you attempt to activate a previously deactivated installation, you will need to use the Block Code to obtain a new Activation Code from my.rationalacoustics.com.
my.rationalacoustics.com	The Rational Acoustics license management site for Smaart registration provides a centralized online location for managing your installations Smaart software license(s) and installations.
Installation	The process of downloading the Smaart installer file and using it to install Smaart on your computer. Installation must be completed before Smaart can be activated.
Activation	The process of license validation. After installation, Smaart will open to the activation wizard, which will present you with online and offline activation options. Each activation will use one installation spot on your license. Note that if you have a virtual machine or multiple operating systems installed on the same machine, each OS that you install Smaart under will each require separate activation.
Deactivation	The process of disabling your Smaart installation to return an installation spot to your license. Deactivation should be performed if a computer is being retired or reformatted, or if the installation limit for your license has been reached and you would like to activate on a different machine. If your

	computer is stolen, please contact support@rationalacoustics.com and we will manually remove (crush) the Machine ID for the stolen machine from your licenses.
Ticket File	A file containing licensing information for a specific computer that Smaart is installed on. The Ticket file can only be read by Smaart's licensing system. Never delete the Ticket file from your computer unless directed to do so either by instructions found in this document or by a Rational Acoustics support technician.

Appendix G: Text File Formats for ASCII Import

Smaart uses plain ASCII text for importing and export of frequency domain data in several different contexts. On the export side, any captured spectrum or transfer function data trace can be copied to the operating system clipboard as an ASCII text table, suitable for pasting into a spreadsheet or saving to a text file for import into other programs. Smaart can also import data from ASCII text files as microphone calibration curves, target curves for banded spectrum displays, weighting curves for all types of frequency domain measurements, or static spectrum or transfer function static data traces (reference traces). With the exception of RTA target curves, the same basic text file formats are used in all cases. Target curves are similar but have a specific header format (see Target Curves on page 86 for details).

The minimum requirement for importing any type of curve is one frequency value in Hertz and one magnitude value in decibels per line, separated by a comma or a tab character – these are commonly referred to as comma-separated values (CSV) or delimited or tab-delimited text. When creating a new transfer function data trace from a text file, Smaart will look for a third column containing phase data in degrees and will import it if found, but this is actually optional. In any case where a text file contains additional columns not required for a given import operation, they are simply ignored. Smaart will also ignore blank lines and any line beginning with a semicolon (;) or an asterisk (*), and so these may be used to add comments, headings or line spaces to help make data files more human-readable.

Comma-Separated Values (CSV)

```
* Free field Microphone Response
* Sensitivity: 6.27 mV/Pa @1kHz

*Freq, Mag (dB)
10.00, -2.22
10.40, -1.90
10.82, -1.84
11.26, -1.73
11.71, -1.51
12.18, -1.43
12.68, -1.47
13.19, -1.28
13.72, -1.08
14.27, -1.05
```

Tab-Delimited Text

```
; Free field Microphone Response
; Sensitivity: 6.27 mV/Pa @1kHz

;Freq	Mag (dB)
10.00	-2.22
10.40	-1.90
10.82	-1.84
11.26	-1.73
11.71	-1.51
12.18	-1.43
12.68	-1.47
13.19	-1.28
13.72	-1.08
14.27	-1.05
```

The two text table formats shown above are functionally identical for import purposes. Smaart will accept either style of formatting for import as weighting curves, microphone calibration curves, or spectrum and transfer function static data traces (reference traces).

Both CSV and tab-delimited ASCII file formats are widely used for exchanging data between many different types of computer programs, meaning that .csv or .txt files from other audio and acoustical programs can often be imported into Smaart with little or no modification. At most, a little bit of hand editing or a search and replace operation in a spreadsheet or text editor may be required.

Please note that Smaart does *not* support commas as decimal points in ASCII data import functions. If you have a data file with fractional values written using commas as decimal points, you will need to replace those with periods before importing that data into Smaart.

Spectrum and Transfer Data Traces

Captured spectrum and transfer function traces in Smaart (often called reference traces) are stored in the form of raw FFT or MTW data, meaning that when you import data from a text file as a stored data trace, you are essentially creating an FFT or an MTW trace. The imported curve can be averaged with other traces of the same type, summed into fractional octave bands (in the case of an RTA trace) or smoothed (transfer function) as you would any other trace. This introduces a couple of potentially complicating factors that you need to keep in mind.

To import frequency domain data from an ASCII text file, make sure the active graph selection matches the type of data that you want to import (spectrum or transfer function) and select *Import* > *Import ASCII* from the *File* menu or click the three-line menu button on the Data Bar and select *Import ASCII*. You will be asked to choose an FFT size and sample rate which will determine the frequency spacing of the new data trace. All frequency data points for the selected FFT size will be created, regardless of whether the file being imported contains data for all frequencies. Smaart will interpolate missing frequency bins from whatever coordinates you supply.

Importing Spectrum Data Traces

When importing a new spectrum data trace from a text file be aware that Smaart spectrum data traces (*.srf* files) are stored un-banded, as linearly spaced FFT data. This means that if you try to import fractional-octave data as a spectrum trace, the resulting curve in Smaart will have an slope of 3 dB per octave, relative to the original. It is *possible* to pre-bias fractional-octave magnitude values prior to importing so that the imported curve sums to the correct fractional-octave values, but hand tuning is typically required and the resulting curve may not work perfectly for every fractional resolution.

If your objective in importing fractional octave banded spectrum data is to plot a target reference curve on the RTA graph, a better option might be to use Smaart's *Target Curves* feature, which simply draws a curve on the fractional octave RTA displays without importing it as an RTA data trace. For more information, please refer to the topic on *Target Curves* in Chapter 4, beginning on page 86.

Importing Transfer Function Data Traces

When importing transfer function data traces from ASCII, Smaart can import magnitude data in decibels and phase data in degrees. Phase is optional. The import function will still work with just frequency and magnitude coordinates.

As with spectrum data imports, you must select a target FFT size (or MTW resolution) for the imported data and any missing data points are interpolated from whatever coordinates you supply. The only caveat is that Smaart 8 does not import coherence data from ASCII files. ASCII exports from Smaart include coherence but Smaart will not import coherence data from ASCII files, due to questions regarding the validity of interpolating coherence values.

```
; Example ASCII Import Format
; For Transfer Function Data
; (Tab-Delimited with Phase)
;
;Freq   →  Mag dB   →  Phase
2.92    →  -10.23   →  -13.31
5.85    →  -11.26   →  -85.54
8.78    →  -8.67    →  -178.93
11.71   →  -2.29    →  85.36
14.64   →  -0.23    →  -4.08
17.57   →  -1.29    →  -56.84
20.50   →  -5.86    →  -144.01
23.43   →  -6.98    →  -65.06
26.36   →  -4.6     →  -49.5
29.29   →  -2.13    →  -108.33
```

Index