

Smaart v.7

User Guide



Rational Acoustics

Smart[®].v.7

User Guide



Rational Acoustics Smart v7 User Guide

Copyright © 2015, Rational Acoustics, LLC. All rights reserved. Except as permitted by the United States Copyright Act of 1976, no part of this book may be reproduced or distributed in any form whatsoever or stored in any publically accessible database or retrieval system without prior written permission from Rational Acoustics, LLC. Smart, SmartLive, Rational Acoustics and the Rational Acoustics logo are registered trademarks of Rational Acoustics, LLC. All other trademarks mentioned in this document are the property of their respective owners.

For information contact:

Rational Acoustics, LLC
241 Church St., Suite H
Putnam, CT 06260
USA

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

Contents

| | |
|--|-----------|
| Introduction | 1 |
| Scope and Purpose of this Guide | 1 |
| What is Smaart? | 1 |
| How to use this Guide | 1 |
| Notation for Accelerator Keys (Hot Keys) and Mouse Clicks | 2 |
| “K” versus “k” | 2 |
| Full Scale (dB FS) versus Full Scale | 2 |
| Recommended Computer Hardware | 3 |
| Loading and Licensing the Software..... | 3 |
| Registering your installation | 3 |
| Chapter 1: Fundamental Concepts and Terminology | 4 |
| Time and Frequency Domain Analysis | 4 |
| Fourier Transforms (DFT/FFT and IFT) | 5 |
| Time Resolution versus Frequency Resolution | 6 |
| Single and Dual-Channel Measurement Techniques | 8 |
| Linear and Logarithmic Scaling..... | 9 |
| Linear and Logarithmic Frequency Scales..... | 10 |
| Linear Amplitude..... | 11 |
| Decibels (dB) | 12 |
| Octave and Fractional-Octave Banding (Spectrum Measurements) | 13 |
| Smoothing (Transfer Function) | 14 |
| Averaging..... | 14 |
| Averaging Over Time (Temporal Averaging) | 15 |
| Spatial Averaging..... | 16 |
| Glossary of Terms..... | 17 |
| Chapter 2: Finding Your Way Around in Smaart | 24 |
| Two Distinct Measurement and Analysis Modes..... | 24 |
| Common User Interface Elements | 24 |
| Graph Area Allocation..... | 24 |
| Active Graph Pane..... | 25 |
| Graph Type Selection | 25 |
| Secondary Graph Controls | 26 |
| Zooming | 26 |

| | |
|--|-----------|
| Cursor Readout | 27 |
| General Options..... | 29 |
| Color Schemes | 30 |
| Managing Configurations | 30 |
| Capturing and Managing Stored Measurement Data (Reference Traces) | 31 |
| Capture Active Measurement | 31 |
| Showing and Hiding Stored Measurements | 32 |
| Organizing your Data with Folders | 32 |
| Averaging Stored Traces | 33 |
| Saving Stored Data Traces to Files for Permanent Storage | 33 |
| Deleting Stored Traces..... | 33 |
| Loading Stored Data from Files into Smart..... | 34 |
| Trace Information (Info) | 35 |
| Weighting Curves..... | 37 |
| Quick Compare | 38 |
| Target Curves | 39 |
| Importing Target Curves | 39 |
| Signal Generator | 41 |
| Pink Noise | 41 |
| Pink Sweep..... | 42 |
| Sine and Dual Sine Waves..... | 43 |
| File-based Signals..... | 43 |
| Chapter 3: Configuring Smart for Real-Time Measurements..... | 44 |
| Configuring Live Measurements | 44 |
| Creating and Managing Live Measurements | 45 |
| Measurement Group Table | 48 |
| Global and Local Measurement Parameters | 48 |
| Managing Audio Input and Output Devices..... | 50 |
| Audio Device Options | 51 |
| Configuring Input and Output Channels..... | 52 |
| Chapter 4: Real-Time Mode User Interface..... | 55 |
| Real-Time Mode Main Window Layout | 55 |
| Cursor Readout | 55 |
| Main Graph Area | 56 |
| Signal Level / Sound Level Meter (or Clock) | 56 |

| | |
|---|-----------|
| Main Display Controls | 57 |
| Signal Generator | 57 |
| Active Measurement Controls | 57 |
| Live Measurement Controls | 58 |
| Chapter 5: Single Channel Measurements | 60 |
| Spectrum Measurement and Display Configuration..... | 60 |
| RTA Measurements..... | 60 |
| Averaging and Banding Controls..... | 61 |
| RTA Graph Types..... | 62 |
| Peak Hold | 62 |
| Plot Calibrated Levels..... | 63 |
| Spectrograph Basics | 64 |
| The Real-Time Spectrograph..... | 65 |
| Spectrograph Dynamic Range | 66 |
| Buffer Size and Slice Height | 66 |
| Grayscale..... | 66 |
| Spectrum Options | 67 |
| Application Examples | 69 |
| Distortion and Overload..... | 69 |
| Feedback Frequency Identification..... | 73 |
| Examining Interaction Patterns with Spectrograph..... | 74 |
| Sound Level Measurement: SPL and LEQ..... | 75 |
| Signal Level / Sound Level (Broadband) Meter..... | 75 |
| Sound Level Options | 76 |
| Sound Level Calibration | 77 |
| Chapter 6: Dual Channel Transfer Function Measurements | 80 |
| Dual Channel Measurement and Display Configuration..... | 81 |
| Transfer Function Control Strip..... | 81 |
| Transfer Function Measurement Configuration | 82 |
| Smoothing..... | 83 |
| Weighting..... | 84 |
| Delay Compensation | 84 |
| Measuring Delays..... | 84 |
| Magnitude Response..... | 86 |

| | |
|---|------------|
| Phase Response | 87 |
| Comparing Phase Traces..... | 90 |
| Unwrapping the Phase Display..... | 91 |
| Phase as Group Delay | 92 |
| Coherence..... | 93 |
| The Coherence Display | 94 |
| Causes of Poor Coherence..... | 94 |
| Live IR..... | 96 |
| Data Protection..... | 97 |
| Transfer Function Options | 98 |
| Application Example: Setting an Equalizer for a Loudspeaker | 101 |
| Chapter 7: Impulse Response Measurement Basics | 105 |
| 1: What is an Impulse Response? | 105 |
| Anatomy of an Acoustical Impulse Response | 106 |
| Propagation Delay | 107 |
| Arrival of Direct Sound..... | 107 |
| Discrete Reflections | 107 |
| Early Decay, Reverberant Build-up, and Reverberant Decay | 107 |
| Noise Floor..... | 107 |
| Uses for impulse response measurement data | 108 |
| Delay Time Measurement | 108 |
| Reflection Analysis..... | 108 |
| Reverberation Time (T60, RT60...)..... | 108 |
| Early Decay Time (EDT)..... | 109 |
| Early-to-late energy ratios | 109 |
| Speech Intelligibility Modeling | 109 |
| Chapter 8: Impulse Response Mode User Interface | 110 |
| Navigation pane..... | 110 |
| Main Graph Area | 111 |
| Cursor Readout..... | 111 |
| Sound Level Meter..... | 112 |
| Data Display Controls | 112 |
| Signal Generator | 112 |
| FFT size and Averaging Controls | 113 |
| Input Source Selector | 113 |

| | |
|---|------------|
| Live Measurement Controls..... | 113 |
| Bandpass Filters | 114 |
| Additional Options for IR Measurement..... | 114 |
| General Options Pertaining to IR Measurement | 114 |
| Impulse Response Options..... | 115 |
| Chapter 9: Analyzing Impulse Response Data..... | 117 |
| Time Domain Analysis | 118 |
| Logarithmic Time Domain Display | 118 |
| Linear Time Domain Display..... | 119 |
| Energy Time Curve (ETC)..... | 119 |
| Bandpass Filtering | 121 |
| Discrete Reflections..... | 121 |
| Reverberation Time..... | 122 |
| Reverse Time Integration | 122 |
| Evaluation Ranges (EDT, T20, T30)..... | 123 |
| Reporting Results for Reverberation Time | 125 |
| Early-to-Late Energy Ratios | 126 |
| Clarity Ratios (C35, C50, C80...)..... | 126 |
| The Histogram Display | 127 |
| The All Bands Table | 127 |
| Frequency Domain Analysis | 128 |
| The Spectrograph | 129 |
| Spectrograph Time and Frequency Resolution..... | 130 |
| Spectrograph Dynamic Range | 131 |
| Spectrograph Analysis of an Acoustical Impulse Response | 131 |
| Speech Intelligibility Metrics | 132 |
| STI and STIPA..... | 132 |
| ALCons..... | 135 |
| Chapter 10: Measuring an Acoustical Impulse Response | 137 |
| What are we measuring, and why? | 137 |
| Direct vs Indirect IR measurement..... | 137 |
| Direct IR Measurement Using an Impulsive Stimulus..... | 138 |
| Indirect (Dual Channel) IR Measurement..... | 138 |
| Dual Channel IR Measurement Using Period-Matched Signals | 139 |
| Dual Channel IR Measurement Using Random Stimulus Signals | 142 |

| | |
|--|------------|
| Selecting Excitation Sources and Positions..... | 143 |
| Directional Loudspeakers and Reverberation Time | 143 |
| Minimum Distance from Sound Sources | 145 |
| Selection of Measurement Positions | 145 |
| Selecting Measurement Parameters | 146 |
| Input source..... | 146 |
| Excitation Level..... | 147 |
| Input Levels..... | 147 |
| Measurement Duration (Time Window) | 147 |
| Averaging and Overlap..... | 148 |
| Pushing the Button and Making the Measurement | 149 |
| Saving Your Work..... | 149 |
| Recap: Recommended Default Settings for Dual-channel IR Measurements | 150 |
| Signal Type..... | 150 |
| FFT size and Averaging..... | 150 |
| Excitation Level..... | 150 |
| Input Levels..... | 150 |
| Delay Time | 150 |
| Appendix A: Applicable Standards and Further Reading | 151 |
| Further Reading on Audio Engineering and Acoustics..... | 151 |
| Applicable Standards for IR Measurements and Speech Intelligibility..... | 151 |
| Appendix B: Room Volume, Absorption and Reverberation time (RT60) | 153 |
| Material Absorption Coefficients..... | 153 |
| Air Absorption..... | 153 |
| Schroeder Cut-off Frequency..... | 154 |
| Practical Measurement Considerations..... | 154 |
| Appendix C: Sound Source Characteristics..... | 156 |
| Directivity Factor (Q)..... | 156 |
| Sound Pressure Level at the Listening Position | 156 |
| Directivity index (DI) | 157 |
| Q and DI figures in Practice..... | 157 |
| Conventional Loudspeaker Arrays | 158 |
| “Line” Arrays | 158 |
| A Note about Echoes | 159 |

| | |
|---|------------|
| Appendix D: Boundary effects | 161 |
| Acoustically Small Sources | 161 |
| Microphones Near Boundaries | 161 |
| Appendix E: Typical Measurement Rig Set-Up | 163 |
| Stereo (2x2) Audio I-O | 163 |
| Multi-Channel I-O | 164 |
| Traditional USB/FireWire Stand-Alone Interfaces | 164 |
| Network Audio I-O | 164 |
| Appendix F: Licensing and Installation | 166 |
| my.RationalAcoustics.com | 166 |
| Registering a License | 167 |
| Installing Smaart v7 | 167 |
| Software Installation on Windows® | 167 |
| Software Installation on Mac OS X® | 167 |
| Activating an Installation..... | 168 |
| Option 1: If this computer is connected to the Internet..... | 168 |
| Option 2: If this computer is not connected to the Internet... .. | 170 |
| Deactivation | 171 |
| Moving Smaart to a New Computer -or- Upgrading your Operating System..... | 171 |
| Using Time Machine, Migration Utilities, or Cloning Software..... | 172 |
| Upgrading your Operating System or Major System Components | 173 |
| Reactivation..... | 173 |
| Reactivating a Smaart Installation on a Deactivated Computer | 173 |
| Reactivating after restoring from a backup or after migrating system files..... | 173 |
| Glossary of Installation and Activation Terms | 174 |
| Appendix G: Text File Formats for ASCII Import | 176 |
| Spectrum and Transfer Data Traces..... | 177 |
| Importing Spectrum Data Traces | 177 |
| Importing Transfer Function Data Traces | 177 |

Table of Figures

| | |
|--|----|
| Figure 1: Dual-channel vs. single-channel measurements in the time domain and frequency domain. | 4 |
| Figure 2: Fourier analysis. | 6 |
| Figure 3: FFT Frequency Resolution shown on a logarithmic frequency scale. | 7 |
| Figure 4: Single-channel vs dual-channel measurements. | 9 |
| Figure 5: Linear vs Logarithmic frequency scaling. | 10 |
| Figure 6: MTW vs 16K FFT transfer function on a logarithmic frequency scale. | 11 |
| Figure 7: Linear vs Logarithmic amplitude scaling. | 12 |
| Figure 8: Fractional-octave banding vs FFT data on a logarithmic frequency scale. | 13 |
| Figure 9: Fractional-octave smoothing for transfer function data. | 14 |
| Figure 10: Real-time mode Averaging selector for the active measurement. | 15 |
| Figure 11: Two distinct operating modes, real-time and impulse response. | 24 |
| Figure 12: Graph Area allocation buttons. | 24 |
| Figure 13: Active graph pane selection. | 25 |
| Figure 14: Graph type selection. | 25 |
| Figure 15: On-graph trace selection menu and threshold control widget. | 26 |
| Figure 16: Mouse zooming (aka rubber band zooming). | 26 |
| Figure 17: Zoom preset options. | 27 |
| Figure 18: The cursor readout shows coordinates for locked and movable cursors. | 28 |
| Figure 19: The General options dialog pane. | 29 |
| Figure 20: The Config Management dialog window. | 31 |
| Figure 21: The data bar. | 31 |
| Figure 22: Quick hide/show captured traces with individual states preserved. | 32 |
| Figure 23: The Trace Info dialog for stored data traces. | 35 |
| Figure 24: The right-click ([Ctrl] + click on Mac) context menu for stored data traces. | 36 |
| Figure 25: ASCII Import dialog for importing trace data from text files. | 36 |
| Figure 26: Custom Weighting Curves dialog. | 38 |
| Figure 27: The Target Curves dialog window. | 39 |
| Figure 28: On-screen signal generator controls. | 41 |

| | |
|--|----|
| Figure 29: Signal Generator output device and channel selection. | 41 |
| Figure 30: Signal generator options for Pink Sweep (log-swept sine) signals..... | 42 |
| Figure 31: Signal generator options for Dual sinewave signals. | 43 |
| Figure 32: Signal generator options for file based signals. | 43 |
| Figure 33: Live transfer function measurement controls on the real-time mode control strip. | 44 |
| Figure 34: Creating a new live spectrum measurement in Measurement Config. | 45 |
| Figure 35: The New Spectrum Measurement dialog. | 46 |
| Figure 36: The New TF (transfer function) Measurement dialog. | 46 |
| Figure 37: Selecting measurements for a live average. | 47 |
| Figure 38: Local and Global measurement settings for a transfer function measurement..... | 50 |
| Figure 39: The Audio Device Option dialog..... | 51 |
| Figure 40: Configuring an audio input device in Smaart. | 53 |
| Figure 41: Anatomy of the main window layout for real-time mode..... | 55 |
| Figure 42: RTA display with Octave, 1/3-octave and 1/12-octave banding..... | 60 |
| Figure 43: Control Strip for Spectrum measurements..... | 61 |
| Figure 44: RTA graph types (Bars, Lines or “Both”). | 62 |
| Figure 45: RTA bar graph with peak hold..... | 62 |
| Figure 46: Turning a spectrum analyzer into a spectrograph. | 64 |
| Figure 47: The real-time mode Spectrograph display..... | 65 |
| Figure 48: Spectrograph dynamic range and color mapping. | 66 |
| Figure 49: The Spectrum options dialog tab. | 67 |
| Figure 50: Loopback setup for distortion and overload example..... | 69 |
| Figure 51: Measurement configuration for distortion and overload example application..... | 70 |
| Figure 52: Signal Generator setup for distortion and overload example application. | 71 |
| Figure 53: Measurement results for distortion and overload application example..... | 72 |
| Figure 54: Measurement system setup for feedback study. | 73 |
| Figure 55: Feedback Study | 73 |
| Figure 56: Spectrograph plot of comb filter interaction patterns. | 74 |
| Figure 57: Sound Level Options dialog..... | 76 |
| Figure 58: Amplitude Calibration dialog for SPL/LEQ calibration. | 77 |

| | |
|--|-----|
| Figure 59: Calibration Progress dialog. | 78 |
| Figure 60: Block diagram of a transfer function and/or dual-channel impulse response measurement. . | 80 |
| Figure 61: Transfer Function measurement control strip. | 81 |
| Figure 62: Transfer function measurement parameters. | 82 |
| Figure 63: Delay Finder button for the active transfer function measurement. | 84 |
| Figure 64: The automated delay finder. | 85 |
| Figure 65: Delay Tracking button for the active transfer function measurement. | 85 |
| Figure 66: The transfer function Magnitude graph. | 86 |
| Figure 67: Uniform (linear) delay on a linear frequency scale. | 87 |
| Figure 68: Uniform (linear) delay on a logarithmic frequency scale. | 88 |
| Figure 69: The standard "wrapped" Phase display. | 89 |
| Figure 70: Phase and magnitude response of a 4th order Linkwitz-Riley bandpass filter. | 89 |
| Figure 71: Comparing phase traces. | 90 |
| Figure 72: Phase on an unwrapped, versus wrapped (normal) phase display. | 91 |
| Figure 73: Phase shown as group delay. | 92 |
| Figure 74: The coherence trace is plotted on scale of 0-100 in the upper half (or quarter) of the magnitude graph. | 93 |
| Figure 75: Coherence blanking. | 94 |
| Figure 76: HF coherence loss in the MTW transfer function due to delay mismatch. | 95 |
| Figure 77: Log and ETC views of the impulse response of a low-frequency bandpass filter. | 96 |
| Figure 78: The Transfer Function options dialog page. | 98 |
| Figure 79: Measurement system setup for Setting an Equalizer for a Loudspeaker. | 101 |
| Figure 80: Creating a new transfer function measurement. | 102 |
| Figure 81: Measurement configuration for EQ measurement. | 102 |
| Figure 82: Loudspeaker measurement. | 103 |
| Figure 83: Transfer function measurement – initial loudspeaker response (Stored), EQ trace (inverted) and equalized loudspeaker response. | 104 |
| Figure 84: Conceptual illustration of an acoustical impulse. | 105 |
| Figure 85: An acoustical impulse response with its common component parts labeled. | 106 |
| Figure 86: The logarithmic (Log) time domain IR graph. | 117 |

| | |
|--|-----|
| Figure 87: 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. | 118 |
| Figure 88: Zoomed in views of the linear impulse response of a bandpass filter, with normal and inverse polarity. | 119 |
| Figure 89: A comparison of the ETC and the impulse response with linear and logarithmic amplitude scaling. | 120 |
| Figure 90: A comparison of the Log IR and ETC graphs | 120 |
| Figure 91: Estimating Reverberation time by reverse integration of the impulse response. | 123 |
| Figure 92: A log IR display with all the bells and whistles. | 124 |
| Figure 93: A scale for interpreting C50 and C80 measurement results for speech and music. | 126 |
| Figure 94: The Histogram graph and All Bands Table. | 127 |
| Figure 95: Moving the time 0 point and selecting a time range for display. | 128 |
| Figure 96: Spectrograph time and frequency resolution as a function of FFT size. | 130 |
| Figure 97: FFT overlap. | 130 |
| Figure 98: The 2K FFT example from Figure 96, shown with 50%, 75% and 99% overlap (left to right). .. | 131 |
| Figure 99: Spectrograph dynamic range and color mapping. | 131 |
| Figure 100: Broadband ETC and Spectrograph of a room impulse response showing a problematic back wall reflection. | 132 |
| Figure 101: Recording of a male voice saying, "Joe took father's shoe bench out." | 133 |
| Figure 102: STI estimates speech intelligibility through a transmission channel as a function of modulation loss. | 134 |
| Figure 103: Block diagram of a Dual-FFT transfer function IR measurement. | 139 |
| Figure 104: Three indirect IR measurements of the same room, taken from the same microphone position using effectively random noise vs period-matched pseudorandom noise. | 140 |
| Figure 105: An impulse response measured using a log swept sine signal (Pink Sweep), showing harmonic distortion products | 141 |
| Figure 106: The effect of averaging on an IR measurement made using random stimulus signals. | 142 |
| Figure 107: Minimum distance to any measurement position from the excitation source (e.g., a loudspeaker) used for reverberation time measurements. | 144 |
| Figure 108: The measurement signal level (M) is running a comfortable level. The reference (R) channel is clipping. | 147 |
| Figure 109: IR Mode FFT size selector showing the time constant in milliseconds for each FFT size. | 148 |

This page intentionally left blank.

Introduction

Scope and Purpose of this Guide

This guide is a practical introduction to configuration measurement and operational concepts in Rational Acoustics' Smaart® v7. Our intent is to provide a comprehensive explanation of the Smaart program, its features and operation, a brief survey of core concepts related to acoustic measurement and analysis and a foundation for making valid, repeatable measurements and extracting some useful information from the results.

Regardless of past measurement experience with previous versions of Smaart, or other analysis systems, you should take the time to familiarize yourself with the process of configuring Smaart version 7. Unlike previous versions, which assumed a simple two-channel I/O, Smaart 7 can interface with multiple I/O devices simultaneously, each with multiple input and output channels. Smaart v7 makes no assumptions about your measurement requirements and upon first run, begins its operation with no preconfigured measurement setups. As in previous versions however, following initial setup, Smaart retains its measurement configuration for subsequent sessions.

This guide assumes that the reader has a basic understanding of professional audio equipment and engineering practices. It 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 several. A list of additional sources of information is provided in the appendices.

What is Smaart?

Smaart is a dual-channel, FFT-based acoustical analysis software platform that enables users 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. The software 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 runs on Microsoft Windows and Mac OS X.

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 almost 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 works with everything from built-in sound chips in laptop and table computers to multi-channel recording interfaces to complex networked digital audio systems.

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 feature, it may not be in the first place you might think to look. We have provided an extensive table of contents and an index to help readers track things

Introduction

down by topic, and of course if you are reading an electronic copy you can also do a full text search. Alternately, you can use the Smaart's online help system, which contains much of the same information and is organized more in parallel with the program itself, to drill down through the menus and dialogs to find what you are looking for.

Notation for Accelerator Keys (Hot Keys) and Mouse Clicks

Since Smaart runs on both Windows and Mac OS X, there are some minor differences in keyboard and mouse commands. 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 keyboard. Also, most PC mice have a two buttons (left and right) whereas many Mac's have only one.

We will write the names of keys used for keyboard commands in 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.

The left mouse button on a Windows machine corresponds to a normal mouse click on Mac – so when we say “left-click,” Mac users just 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.

Summary of notational conventions for keyboard and mouse operations

[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.

A 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.

“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 or 1024 (2^{10}), depending on the context. In this guide, we will strive to adhere to the new SI convention of using the lower case “k” to denote *only* multiples of 1000, and 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 Hz and 8K as meaning 8192 (8×1024).

Full Scale (dB FS) versus Full Scale

There now exist two competing references for decibels in digital audio signals. The first 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. 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 level of a full-scale digital sinewave is -3.01 dB FS.

Recommended Computer Hardware

While Smaart v7 will operate on a wide range of computer hardware configurations, we recommend the following minimum computer configuration for new installations:

Windows®

- Operating System: Microsoft Windows XP or later
- CPU: 2 GHz or faster dual-core Intel architecture (or compatible) processor
- RAM: 2 GB or greater
- Video: Graphics processor with at least 128 MB dedicated video RAM, minimum 1024 x 600 pixel display
- Sound Hardware: Audio Hardware with OS compatible ASIO, Wav/WDM drivers.

Macintosh

- Operating System: Mac OSX 10.5, 10.6, 10.7, or 10.8 (Leopard, Snow Leopard, Lion, Mountain Lion)
- CPU: 2 GHz or faster dual-core Intel architecture processor
- RAM: 2 GB or greater
- Video: Graphics processor with at least 128 MB dedicated video RAM, minimum 1024 x 600 pixel display
- Sound Hardware: Audio Hardware with OS compatible CoreAudio drivers.

Loading and Licensing the Software

To install Smaart v7 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 initial installation, the first time you run Smaart you are presented with an activation screen containing your computer’s 14-digit “Machine ID” and prompted for an “Activation Code” to authorize the installation. The Machine ID is a unique code that identifies the particular computer that you are installing. The Activation Code is a unique code that will only work with that Machine ID.

If you are installing Smaart on a computer that is connected to the internet, you can choose the “Activate Online Now” option to acquire the necessary code and activate the installation. Otherwise, make note of the Machine ID and log into <http://my.RationalAcoustics.com> from another computer to acquire the activation code for the target machine. For more information on installation, registration and license management, please refer to *Appendix F: Licensing and Installation*.

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 v7'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 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.

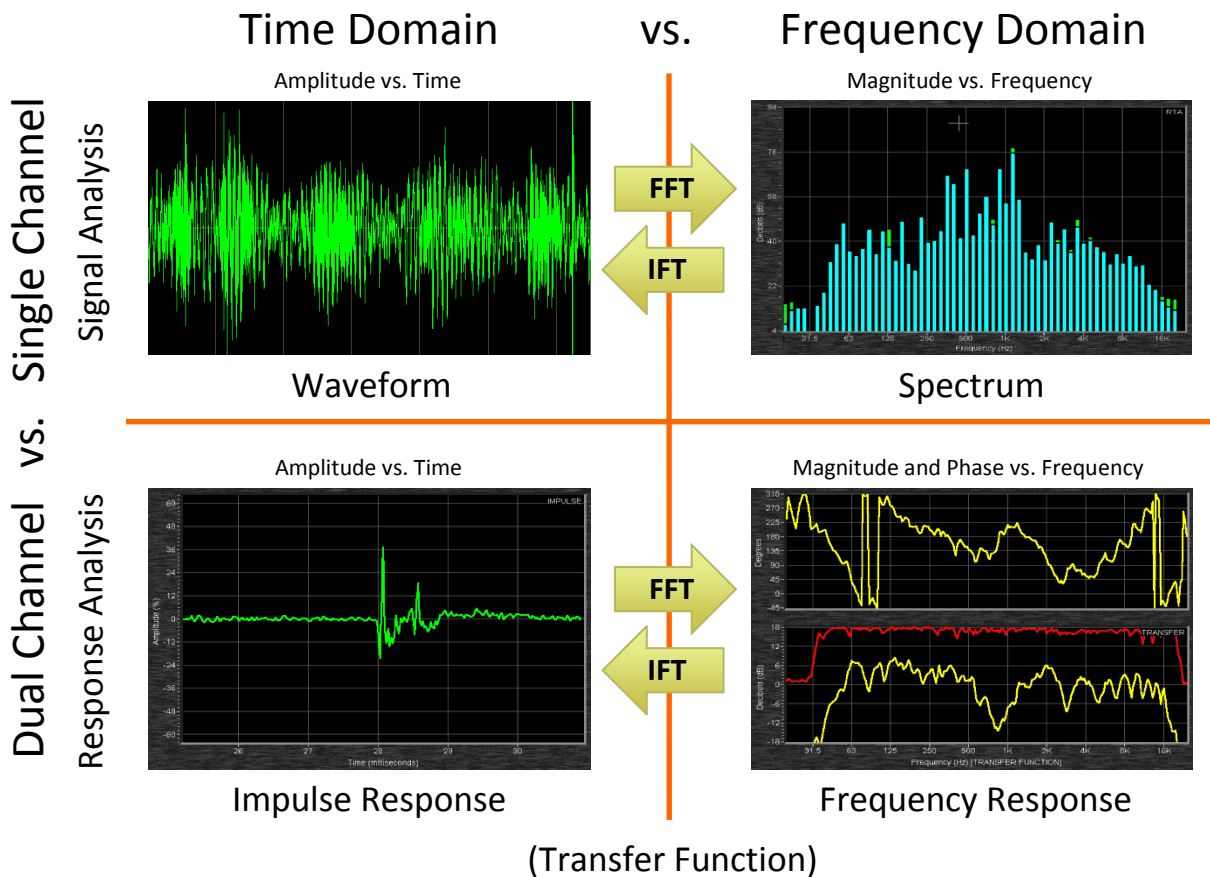


Figure 1: Dual-channel vs. single-channel measurements in the time domain and frequency domain.

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 do 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 other application where the accompanying restrictions on the length of the time record are not a problem.

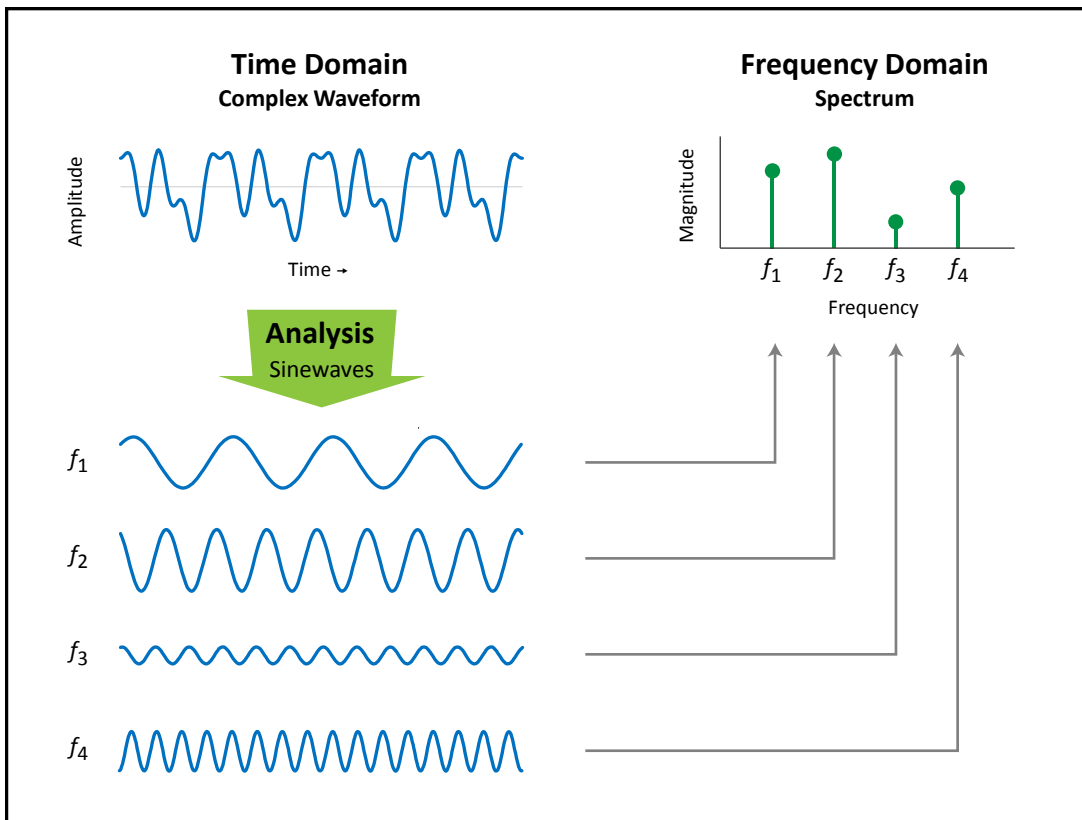


Figure 2: Fourier analysis. The discrete Fourier transform (DFT or FFT) analyzes a complex time-domain signal to find the magnitude and phase of the component sinewaves that make up the complex waveform. The magnitude of each component sinewave can be plotted on a frequency-domain graph to form a picture 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 or 341 milliseconds (16384 / 48000).

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

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).

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

If you are familiar with the reciprocal relationship between cycle time and frequency in sinewaves ($f = 1/t$ 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, so knowing the time constant also tells you how far apart the frequency bins are.

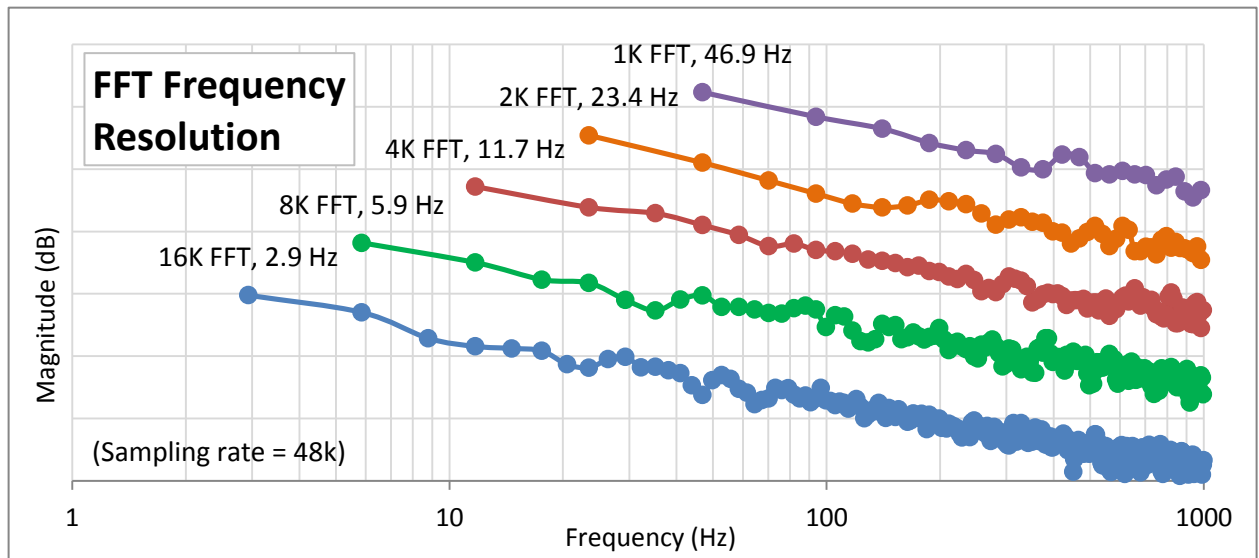


Figure 3: FFT Frequency Resolution shown on a logarithmic frequency scale. Each doubling of FFT size (in samples) doubles the FFT frequency resolution and extends its frequency range an octave lower.

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 analysis of 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 ($60 / 0.341 \approx 176$). 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.

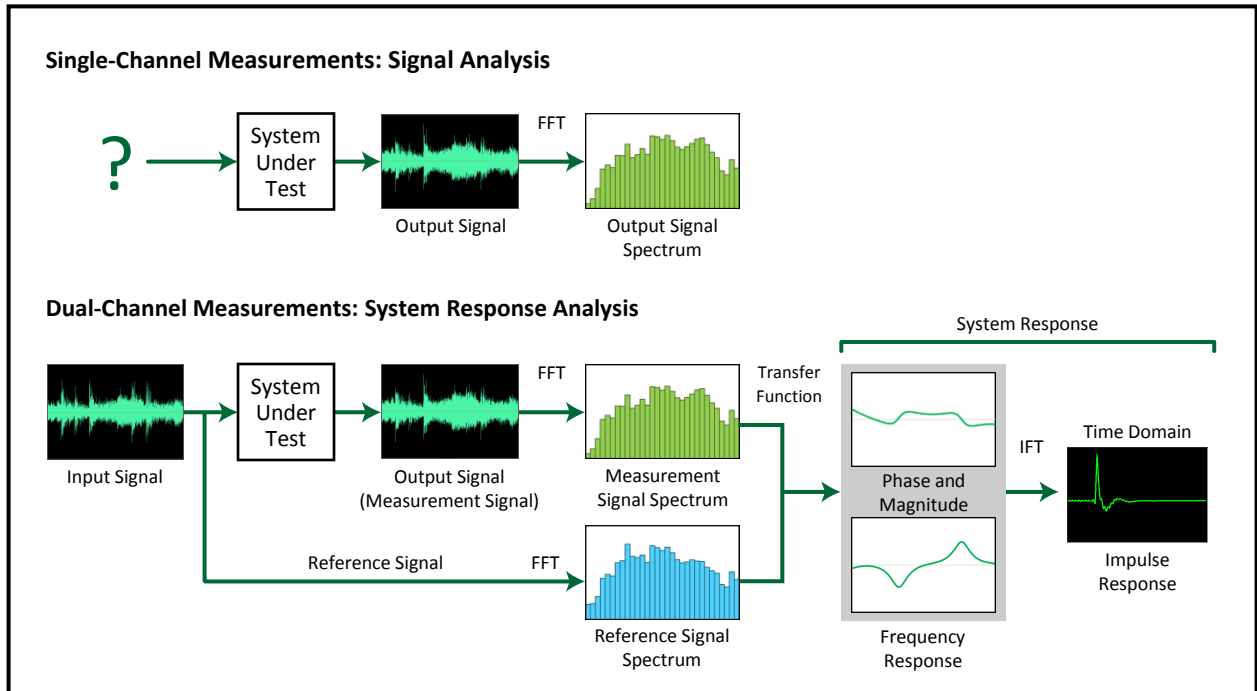


Figure 4: Single-channel vs dual-channel measurements. Single channel spectral measurements analyze the energy content of time domain signals. If the signal being analyzed is the output of a sound system and you happen to know the spectrum of the input signal, you can infer an estimation of the system’s 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) which 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 of us, 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 don't 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) and/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 Smart, 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 Smart are two and ten (octaves and decades).

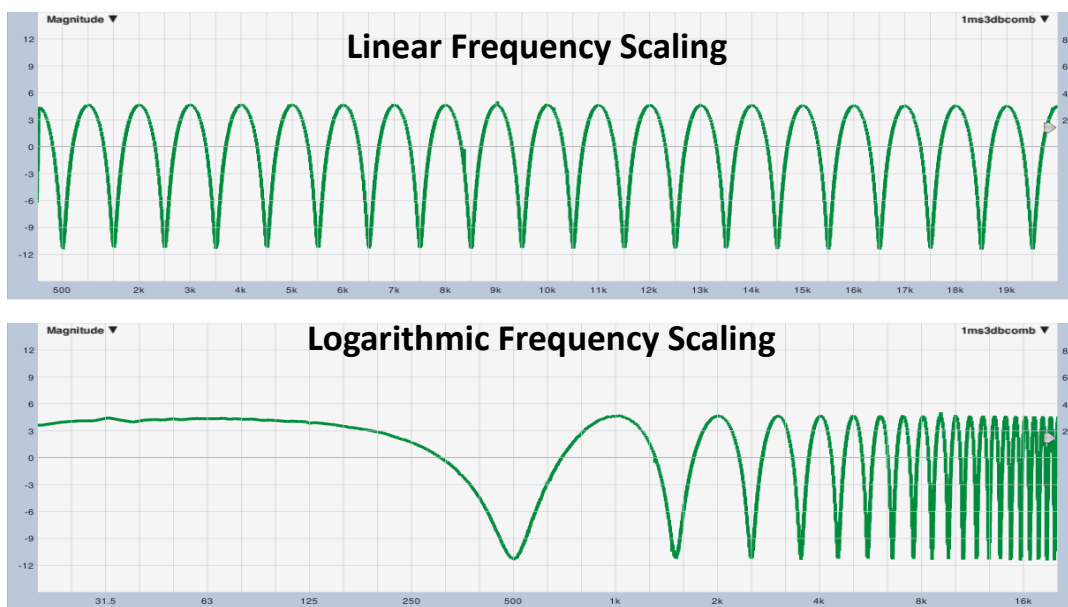


Figure 5: Linear vs Logarithmic frequency scaling. Two views of the same comb filter on a linear and log scaled magnitude graph.

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, 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.

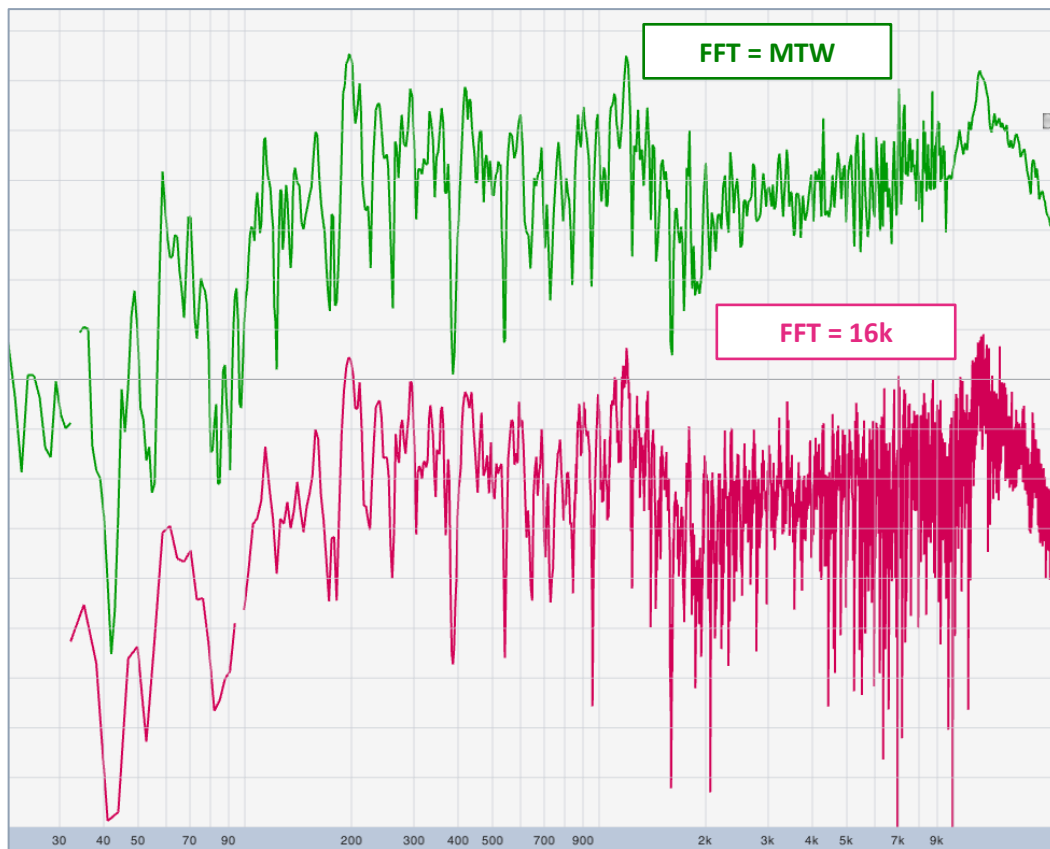


Figure 6: MTW vs 16K FFT transfer function on a logarithmic frequency scale. The MTW uses larger time constants at low frequencies to improve LF resolution while smaller time constants at higher frequencies reduce visual “noise” due to excess resolution.

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 is on 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 are scaled to a range between 1 and -1 (inclusive), with fractional values 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 waveform period is spread out of time so much that a clear impulse is not easily discernable.

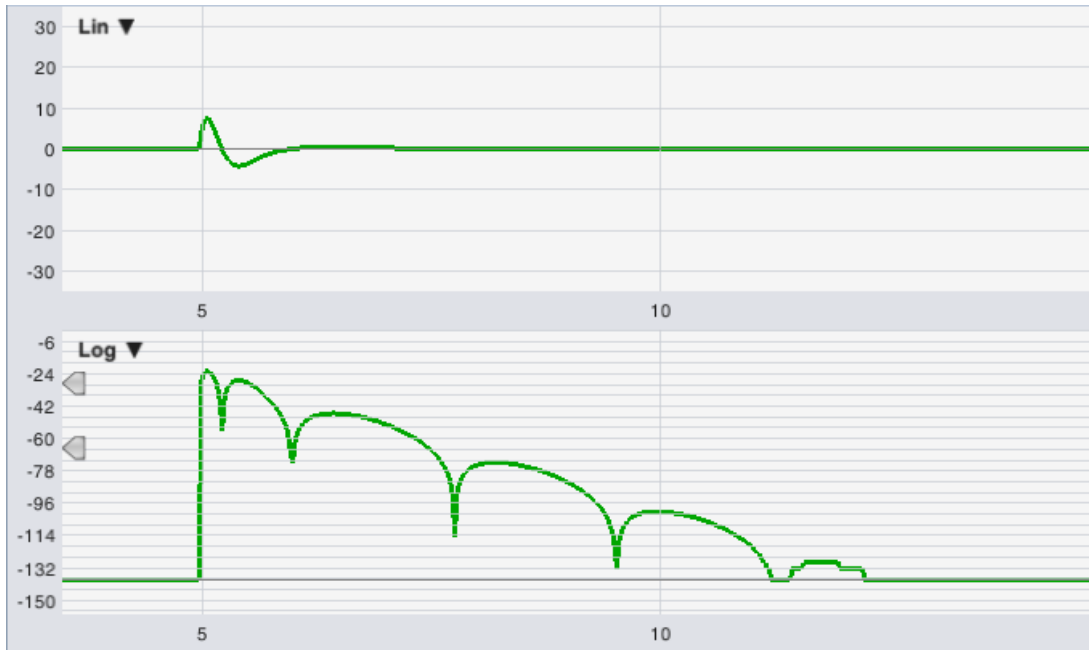


Figure 7: Linear vs Logarithmic amplitude scaling. The impulse response of a bandpass filter is shown on a linear (percentage of normalized full scale) versus logarithmic (decibel) amplitude scale. Notice that only the first two oscillations in the IR are easily discernable on the linear (Lin) view, 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 above, the reference is assumed to be one, however it could potentially be any other number. To reference dB to a number other than one, you can 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 in audio applications is dBu, which references 0 dB to 0.775 Volts. Another is 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 appear flat on an unbanded linear or logarithmic spectrum display, but slopes upward at 3 dB / octave on a banded display.

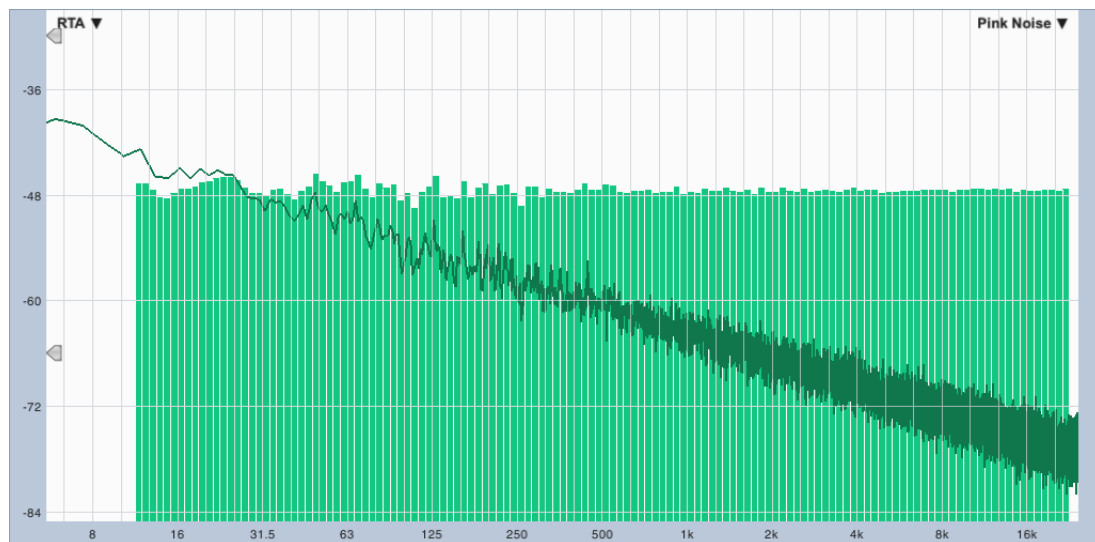


Figure 8: Fractional-octave banding vs FFT data on a logarithmic frequency scale. The lighter green bars show a 1/12-octave banded RTA measurement of pink noise. The darker green line trace is a narrowband (un-banded) view of the same data on a logarithmic frequency scale.

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 numerous studies have found long-term average spectra for all kinds of music to be similar to that of pink noise, across a range of genres and cultures. 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 filtering out small fluctuations in magnitude and phase to help make larger features and trends in the data easier to see. It is analogous to fractional octave banding except that it works by averaging instead of summing. Smoothing places each frequency data point at the center of its own “band,” rather than collecting the bins into a fixed number of bands at fractional octave intervals. In this case, each “band” is a center-weighted smoothing window that expands as you ascend in frequency.

On a logarithmic display, as the linearly spaced frequency FFT data points get closer and closer together at high frequencies, the smoothing window

opens up to include more points on either side in the average. This helps to tame the HF “fuzziness” inherent in plotting FFT-based measurements of noisy signals on a logarithmic frequency scale.

Averaging

Averaging is used a number of different ways in Smart, to try and separate useful information from extraneous factors such as noise, reverberation and position-dependent acoustical anomalies. Averaging in Smart 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 9: Fractional-octave smoothing for transfer function data.

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 in both cases 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 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 strip that runs down the right side of the main window (see Figure 10). 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 (sometimes 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.

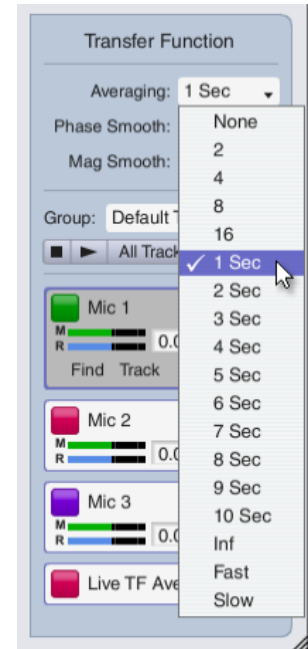


Figure 10: Real-time mode Averaging selector for the active measurement.

Chapter 1: Fundamental Concepts and Terminology

- *Fast* and *Slow* averaging models the decay characteristics of standard Fast and Slow exponential time integration used in standard sound level meters.
- 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 7 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’s easy to compare them in real time to see which one seems to give you a better answer 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 are averaging measurements taken at different locations, rather measurements made from a single location at different times. Spatial averaging is 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 overall coverage from a loudspeaker.

If you have multiple microphones and inputs available, you can do spatial averaging in real time. It can also be done by averaging stored measurement snapshots captured at different locations. We will cover live measurement configuration and working with stored data snapshots in the next section. The following options apply to both methods.

Power vs Decibel Averaging (RTA Measurements)

For RTA measurements, you have the option of power versus decibel averaging in spatial averages. 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. Power averaging gives more weight to the loudest sounds in each frequency and may result in a graph that looks like it sounds.

Decibel averaging is a simple arithmetic averaging of decibel magnitude values may be preferred for sound system measurements in general. You might say that it gives you more of a “consensus” view than power averaging. For example, if one of the measurements in a power average comes in at a significantly higher level than all the others, it will dominate the average and could change the shape of the entire curve significantly. In a decibel average, the higher-level measurement would move the level of the overall averaged curve higher but would not affect its shape significantly more than any other contributor to the average does.

Coherence Weighting (Transfer Function Measurements)

For transfer function data, decibel averaging is used for magnitude data and phase averaging is based on complex data. You do however have the option of using coherence weighting (or not). Coherence weighting gives more weight in the average 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 – 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 a localized reverberant buildup, or perhaps it was taken near the edge of the coverage pattern of a loudspeaker where the HF was rolling off, coherence weighting should result in the more trustworthy frequencies contributing more to the average than problematic frequencies.

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 7 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 Discrete 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 on 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 peak signal value and all lesser decibel values are negative. A second convention preferred by the Audio Engineering Society (AES) references 0 dB to the RMS value of a full-scale peak-to-peak sinewave (i.e., 0.7071 rather than 1.0). In Smaart 7, 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 response of a system to an impulsive stimulus in the time domain. The impulse response of linear time invariant (LTI) system is also the inverse Fourier transform of its 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 7 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 histogram (bar chart), rather than a line trace or scatter plot, 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 $360^\circ =$ 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 1kHz 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 \approx 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 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 Smart

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 strip, or via the *Mode* menu, or by using the [I] and [R] hot keys. Yet another way to do it is by recalling a view preset, based on one mode or the other, but perhaps we are getting ahead of ourselves.

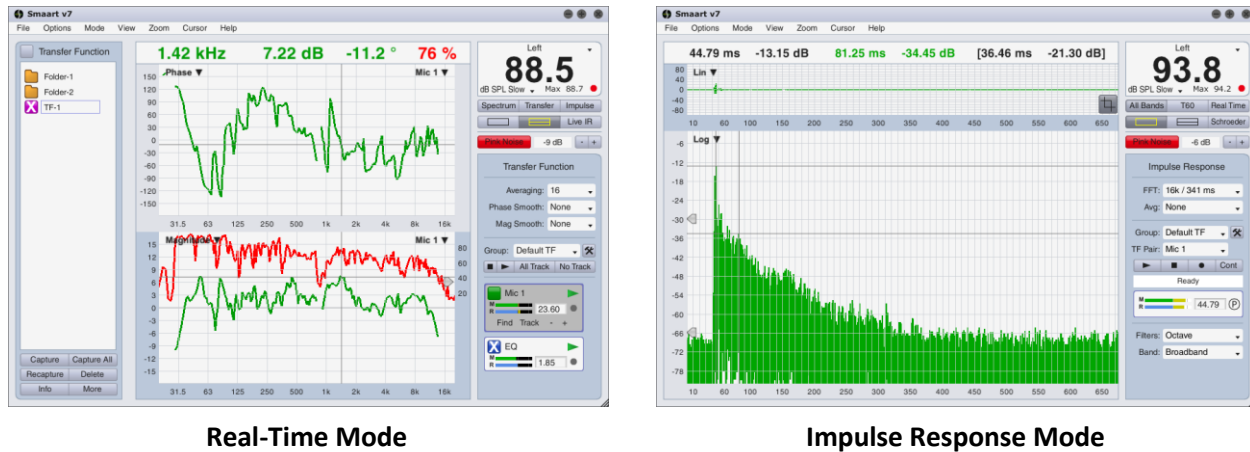


Figure 11: Two distinct operating modes, real-time and impulse response.

Both modes 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. It is optimized for onsite 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. Smaart 7’s impulse response now includes most of the functionality from the old Smaart AcousticTools software package including analysis of reverberation times, early to late energy ratios and speech intelligibility metrics.

Common User Interface Elements

At first glance, the main window layouts for real-time and impulse modes look very similar. Each has a “Control Strip” on the right side of the main window with a large numeric signal level/sound level meter above. To the left, is the main graph area with the cursor tracking readout above it, which takes up the bulk of the window area. Real-Time mode also has a Data Storage area on the left side of the window, but that can actually be hidden by selecting Toggle Data Bar from the Options menu or pressing the [B] key on your keyboard.

Graph Area Allocation

The main Graph area in both real-time and impulse modes can be subdivided into multiple panes. In real-time mode, the main graph

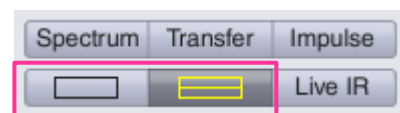


Figure 12: Graph Area allocation buttons.

area can be divided into two panes, 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 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, that is used for navigation purposes and is always visible. The lower portion of the graph area can 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 and Spectrograph).

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 are present, one of them is always considered to be the *active* display. 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. The margins of the active graph are colored to match the background color of the control strip on the right side of the main window, indicating that these two things are connected.

In real-time mode the active graph pane determines which controls you see in the control strip and which set of stored data you see in the data bar on left the side of the window (Spectrum or Transfer Function). It is also the source for data capture operations. The active pane is also the focus of any keyboard or menu commands that affect what you see on a graph, such as hot-key zooming or cycling the z-order of the data traces.

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 13: Active graph pane selection.

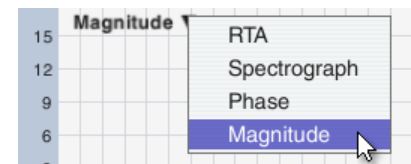


Figure 14: Graph type selection.

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.

On-Graph Menus

In real-time mode, all graph panes have a menu in the upper right corner that displays the name of the trace that is currently on top. Clicking on this label pops up a menu showing you the names of all traces on the graph – both live and captured measurements – arranged in their current z-axis (front to back) stacking order. Clicking the name of any trace in this list with your mouse brings it to the front of the stack and moves its name to the top of the list. Note that clicking on any visible live measurement in the control strip to the right of the graph or on a visible captured trace in the data bar on the left will also bring that to the top of the z order. Additionally, you can step through the z order on real-time charts by pressing the [Z] key on your keyboard.

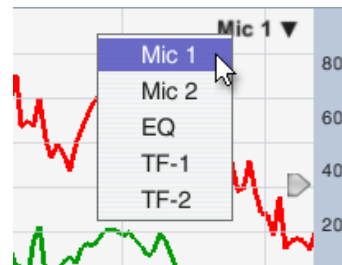


Figure 15: On-graph trace selection menu and threshold control widget.

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.

Thresholds

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 (see Figure 15).

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. If you have a two-button (or more) mouse or track-pad, you can right-click (or more) mouse or track-pad, you can right-click and drag to draw a “rubber-band box” around the area you want to zoom in on, then release the mouse button. If you have a Mac with a one-button pointing device, hold down the [Ctrl] key while clicking and dragging. Either way, when you release the mouse

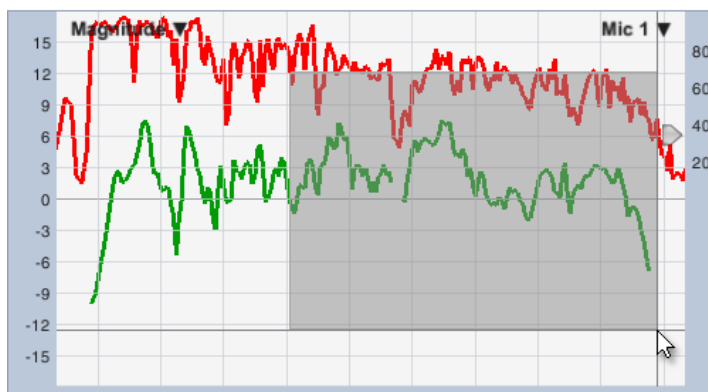


Figure 16: Mouse zooming (aka rubber band zooming). Right-click ([Ctrl]+click on Mac) and drag with your mouse on any graph to select a zoom range for display.

button the selected area expands to fill the full graph area. Clicking in the margin of any plot that you have zoomed in on restores it to its default zoom range.

The plus and minus keys ([+] / [-]) can be used to zoom in any out on the vertical axis of any graph in Smaart except the small 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 axis of the active graph. [Ctrl/Cmd] + [+/-] zooms in/out on the both the x and y axes of the active graph.

Zoom presets enable you to zoom the x axis of a graph to predefined ranges or in preset increments, depending on the graph type.

- On frequency domain graphs, selecting *Zoom 1-4* from the *Zoom* menu or pressing [Alt/Option] + [1-4] on your keyboard sets the sets the x-axis of the active graph to the corresponding frequency range defined in the *Frequency* section of on the *Zoom* options page (*Options* menu > *Zoom*).
- The second set of zoom presets (*Absolute Time*) on the *Zoom* options page applies to time-domain charts in impulse response mode.
- On the transfer function Live IR display, the *Relative Time* zoom presets work as coarse and fine zoom in/out commands, similar to [Alt/Option] + [+/-] but in user-definable increments, as specified in the *Relative Time* section of the *Zoom* options page.
- 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.

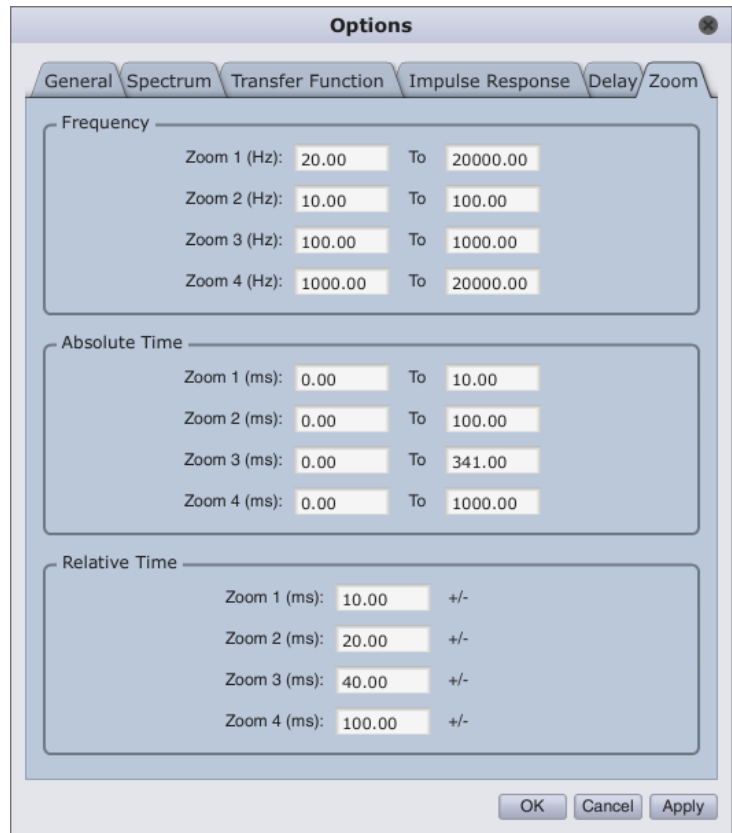


Figure 17: Zoom preset options.

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 from the *Options* menu. (*Options > General*).

Mouse Cursor Tracks Data (or Not)

As you move your mouse cursor over any graph in Smaart, assuming there is data visible on the graph, you will notice a set of “cross hairs” – a horizontal and vertical line extending all the way across the plot from left to right and top to bottom – following the mouse cursor as you move it around. The coordinates for the point where these two line cross are reported in the numeric cursor readout. 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 but snap to the nearest amplitude/magnitude data value on the vertical axis. If multiple traces are present the cursor tracks the top trace in the z order.

You can turn off data tracking for the movable cursor by unselecting *Free Cursor Tracks Data* in the 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.

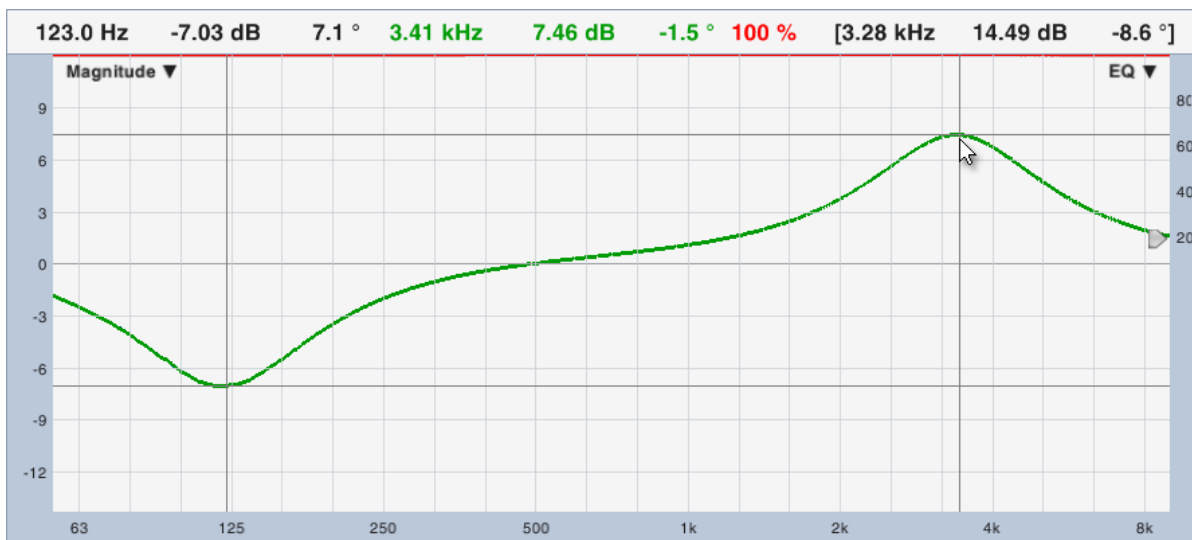


Figure 18: The cursor readout shows coordinates for locked and movable cursors.

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 being 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 *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 Smart's appearance and behavior in both real-time and impulse response mode.

API

If you are sharing Smart measurement data with a compatible client application, the publisher of the client application should provide instructions for the settings in this section.

- The *Enabled* button turns on data sharing and begins broadcasting a “here I am” message on the local network. This option should be turned off if not in use.
- The *Port* field sets the port number for the network socket address.
- The *Connected* indicator turns from red to green when a client application connects.

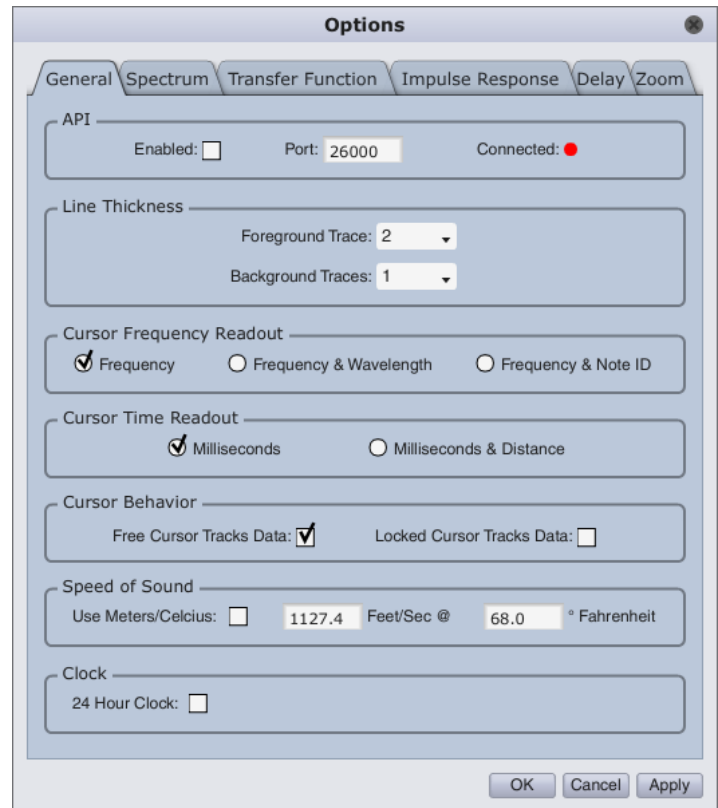


Figure 19: The General options dialog pane.

Line Thickness

The *Line Thickness* controls set the line thickness for all line charts in Smart, 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

The options in this section control 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.

Chapter 2: Finding Your Way Around in Smaart

- *Frequency and Wavelength* – cursor readout displays frequency in Hertz and the wavelength in feet or meters, based on time and temperature units selection in the *Speed of Sound* section.
- *Frequency and Note ID* – cursor readout displays frequency in Hertz and the closest musical note corresponding to that frequency.

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 in the upper right corner of the main window when you press the [K] key on your keyboard.

Color Schemes

Smaart has two built in color schemes, the default light-on-dark scheme that you see the first time you run the program and a higher-contrast scheme with darker-colored text and data traces on a lighter background. The default scheme works well for indoor work, particularly in darkened rooms. The high-contrast scheme may be a better choice for working outdoors in daylight, in brightly lit rooms, or when making screen shots for print documents or slide presentations. To switch to the high contrast color scheme, select *High Contrast* from the *Options* menu. Note that a check mark appears in the menu next to the name when this option is selected. Selecting the same command again turns it off.

Managing Configurations

Nearly all of Smaart's user-configurable settings are stored in a file so that Smaart can "remember" them when you shut down the program. They also provide an easy way to switch between setups for

different tasks or work environments. Configuration (config) files are stored in a private location and managed through the *Config Management* dialog, accessible from the File menu (*File menu > 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.

The current configuration includes all of your audio device selections including friendly names and calibration settings, measurement setups, 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 that setup in the future. To store a configuration, open the configuration manager and click the *Save As* button in the *Current Config* section. You will be prompted to name the new configuration. When you have done that, 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.

Note that config files do not include stored measurement traces or their display settings. Those are stored and managed separately, so changing configurations does not affect your stored data. Additionally, a few parameters are intentionally omitted from the config file because we have found it best to reset those to their default states each time the program starts up.

Capturing and Managing Stored Measurement Data (Reference Traces)

In real-time mode, the Data Bar on left side of the main window in Smaart is dedicated to storing and managing static “snapshots” of live data traces. (Long-time Smaart users know these as reference traces.) The Data Bar can be hidden to make the graph area larger, so if you do not see a column of controls on the left side of your Smaart window in real-time mode, press the [B] key on your keyboard (or *select Toggle Data Bar* in the *Options* menu) to display it before reading on.

Capture Active Measurement

To capture a snapshot of any live real-time measurement in Smaart, make sure the measurement that you want to capture is the active measurement on the

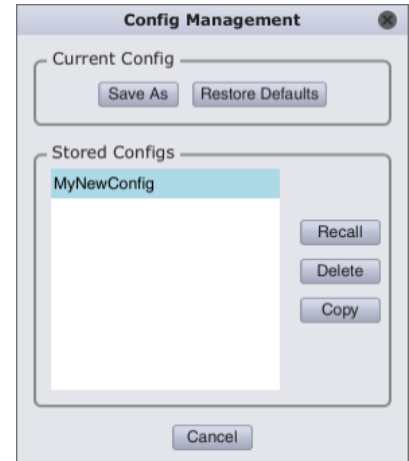


Figure 20: The *Config Management* dialog window.

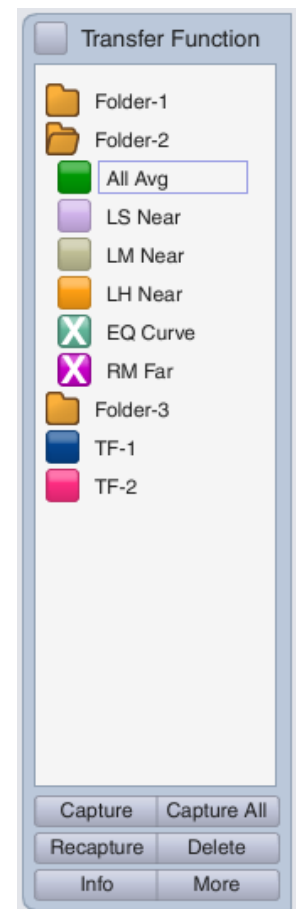


Figure 21: The data bar.

active graph, then press the spacebar on your keyboard or click the *Capture* button, located in the lower portion of the data bar. A small dialog box pops up where you can type a name for the captured trace (a default name is suggested that you can simply accept if you like). The new captured trace will appear in the data bar when you click OK.

Recapture

Clicking the *Recapture* button performs the same operation as a capture (see above), except that instead of creating a new captured trace, you will be replacing the contents of the previously captured trace that is currently selected in the data bar. In this case, you are not asked for a name, since the existing trace that you are replacing presumably already has one.

Capture All

Capture All ([Shift] + [Spacebar]) is a batch operation that captures every live measurement visible on the active graph. Once again, you are asked for a name. When you have decided on one, and click *OK* in the pop-up window, Smart will create a folder in the Data Bar with the name that you specified and place all the captured measurements in it. Each measurement has the same name as the folder, with a number appended at the end. The colors of the captured traces will be set to match the live measurements they were captured from, so that you can tell which is which by color.

Showing and Hiding Stored Measurements

To hide a stored data trace that you don't want to see on your screen for some reason but don't want to delete, simply click on its color tile in the data manager with your mouse. A large letter "X" will appear in the color tile to indicate the trace is hidden. Click on the tile again removes the X and restores the trace data to applicable charts in the graph area.

Hide All Stored Measurements

If you want to hide all stored measurements there are two ways to do it. One is to click the gray button at the top of the data bar, next to the active data set label (Spectrum or Transfer Function). Doing this removes all stored data traces from applicable charts but does not disturb their individual show/hide states. The other way to do it is to click on the *More* button at the bottom of the data bar and select *Hide All* from the pop-up menu. Unlike the first method, this is like going through every trace in the data bar and hiding each one individually. There is no un-hide all method if you use the second method. Instead, you have to click the color tile for each trace that you want to show again with your mouse to unhide it.

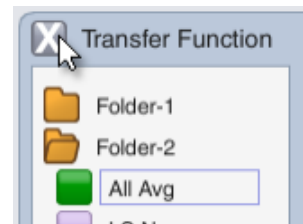


Figure 22: Quick hide/show captured traces with individual states preserved.

Organizing your Data with Folders

When you have a lot of captured traces, it may be helpful to use folders to organize your data. To create a new folder, click the *More* button at the bottom of the data bar and select *New Folder* from the pop-up menu. A new folder is created with the default name "Folder" and a number. You can double-click on the name to edit it, then press the [Enter] key on your keyboard to set the change.

To add a stored trace to the folder, click on its name with your mouse and drag it over the folder icon, then release the mouse button to drop it into the folder. You can also drag files out of folders to other positions in the list. Once you have some traces in your folder, you can click on the folder icon to collapse it and click again to expand the contents.

Averaging Stored Traces

Averaging stored traces is one way of doing spatial averages without using multiple microphones and input channels. Smaart makes this easy to do. To create a stored measurement trace that represents the average of a group of other stored measurements, click the *More* button at the bottom of the data bar and select *Average* from the pop-up menu. This opens the Trace Average dialog window, where you can select the traces that you want to contribute to your average trace, and set the trace name and color for the averaged trace. Note that all traces going into the average need to have the same FFT size and sampling rate.

When averaging spectrum traces, you have the choice of averaging magnitudes as decibels (*dB*) or *Power* (squared linear magnitude) – most often, power averaging is used for spectrum data but there are exceptions to every rule. With transfer function averages, you have the option of creating a *Coherence Weighted* average that uses the coherence value of each frequency data point in each trace as a weighting factor, rather than giving all data points from all traces the same relative weight in the average. The better the coherence, the more weight is given.

Saving Stored Data Traces to Files for Permanent Storage

To write any captured trace to a data file for permanent storage, select the desired trace object in the data bar, click the *More* button and select *Save* from the pop-up menu or press [Ctrl/Command] + [S] on your keyboard. You will be prompted to choose the destination folder and a file name for the new file. The extension *.srf* (spectrum reference file) or *.trf* (transfer function reference file) is appended to the file name, depending on the type of measurement you are saving.

Save Many

To save a group of files to a permanent storage location, click the *More* button and select *Save Many*. This will open a dialog window where you can select which traces you want to save by clicking the check boxes next to each trace name. In this case, the names of the traces are used for their file names. You are not asked to name them individually. Notice that when you click the check box next to a folder name, all the files within it are automatically selected. Smaart will create a folder of the same name in the destination folder and write those files into it. When you click *Save*, all selected traces are written to the location shown in the *Path* field. To change it, you can click the *Browse* button to navigate to a different location or create a new folder. The *Delete register(s) after save* option deletes all selected files and folders from temporary storage and removes them from the Data Bar when the files are saved.

Deleting Stored Traces

Captured traces shown in the data bar in Smaart are written to temporary files in a private location upon capture, and automatically reloaded into memory each time you run the Smaart. Deleting a trace from the Data bar removes the corresponding temporary data file and frees up memory. It is a good

practice to clear out unneeded data traces from the data bar regularly, after backing up any that you want to keep for posterity using the *Save* or *Save Many* functions. Doing so frees up RAM, keeps the data bar manageable, and decreases the time it takes to change banding settings for spectrum data.

To delete a single captured trace from the data bar, just click on its name to select it and then click the *Delete* button in the lower portion of the data bar or press the delete [Del] key on your keyboard. If you have multiple traces to delete, you can also use the *Delete Many* function.

Delete Many

If you want to delete several stored traces at once click the *More* button at the bottom of the data bar and select *Delete Many* from the pop-up menu. A dialog box will appear with a list of all stored traces currently loaded in the data bar. Next to each file and folder name is a check box. You can select the files that you want to delete by clicking these check boxes with your mouse. If you select the check box for a folder, the folder and all of its contents will be deleted. If you want to delete just some of the files in a folder, you can click its icon to expand the contents and then select files individually. When you have finished making your selections, click the *Delete* button to proceed with deleting the selected traces.

Loading Stored Data from Files into Smaart

To load data traces stored in files into Smaart, first you need to make sure the active display type matches the type of data file that you want to load. The active display (*Spectrum* or *Transfer Function*) type is indicated by the headings at the top of the data bar and the control strip.

If you want to load spectrum data files stored in Smaart 7 .srf files, make sure the heading at the top of the data bar says "*Spectrum*." If it doesn't, then switch the active display to an *RTA* or *Spectrograph* to bring up the spectrum data bar before continuing. If you want to load in transfer function (.trf) data files, the active display must be a *Magnitude*, *Phase* or *Live IR* graph and the data bar should say "*Transfer Function*" at the top.

Once you have the correct version of the data bar loaded, click the *More* button at the bottom of the data bar and select *Load* from the pop-up menu. This will open a file system dialog, wherein you navigate to the folder where the files that you want reside and open them. You can select multiple files from the same folder by holding down the [Ctrl/Cmd] key on your keyboard while clicking on the filenames, then click the *Open* button to load all selected files into Smaart.

Note that in addition to its own .srf and .trf files, Smaart 7 can open .ref files created by older versions of Smaart all the way back to version 1.0. Smaart 7 cannot however, open reference group (.rgp) files from older versions of Smaart Pro or SmaartLive. If you have data stored in an .rgp file that you want to open in Smaart 7, you will need to open it in a compatible version of SmaartLive or Smaart Pro and save the traces that you want to open in Smaart 7 as individual reference trace (.ref) files.

Load Folder

Another way of loading multiple trace data files into Smaart is to use the *Load Folder* function. To load all compatible files contained in any folder in your computer's file system, click the *More* button at the bottom of the data bar and select *Load Folder* from the pop-up menu. A dialog box opens with a tree view of your file system wherein you can navigate to the folder you want to load, then click the *OK*

button to load it. Smaart will create a folder of the same name and load all compatible files in the folder, based on the current active display type. If the Spectrum data bar is active, all .srf file and any .ref files containing spectrum data are loaded. Loading a folder from the *Transfer Function* data bar imports all .trf files and data from any .ref files in the folder that contain transfer function data.

Trace Information (Info)

Notice that when you hover over the name of a stored trace in the data bar, Smaart pops up a little window (called a “tool tip”) that shows you the full name of the trace, the date and time that it was created, and the names of the live measurement and measurement group that it was captured from (if known). To see more detailed information about a trace or change its display parameters, you can select its name in the data bar and click the *Info* button, or right-click ([Ctrl]+click on Mac) on its name and select *Info* from the pop-up menu. Either action opens the *Trace Info* dialog window for the selected trace.

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

Below the vital statistics list are editable text fields for the trace name and an optional comment. Next comes the display color for the trace. You can change the color if you like, by clicking on the color tile to open the 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 area when applied (i.e., by clicking the *Apply* button or clicking *OK* to apply changes and exit the dialog). Positive values move the trace up on the chart by the specified number of decibels. Negative numbers move it down. Note that dB offsets are not stored in Smaart’s config file, meaning that they get zeroed out when you close the program and restart it.

The *Weighting* control assigns a weighting curve to the trace. Smaart has built-in curves for A, B and C weight curves used in SPL and LEQ measurements and the cinematic X curve, 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 37).

The *Plot* control sets the preferred plot assignment for the trace when two charts of the same type are displayed in the main graph area. If this is set to 1 (the default setting for all traces), the trace will appear in the first of the two charts to be 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.

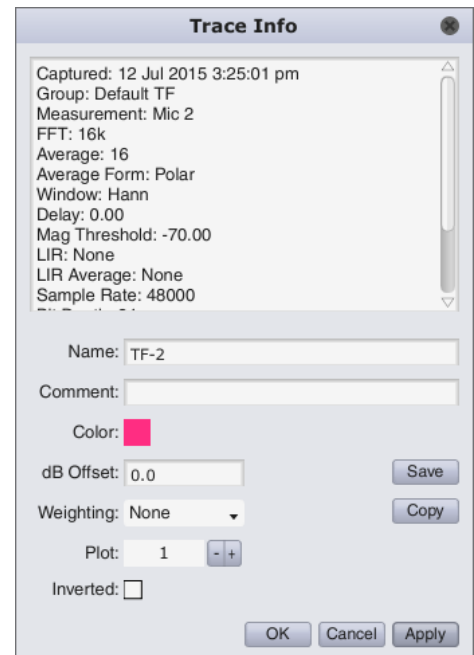


Figure 23: The *Trace Info* dialog for stored data traces.

The *Save* button in the *Trace Info* dialog does the same thing as the *Save* command in the *More* menu. It opens a file dialog window, enabling you to save the trace to a data file for permanent storage.

The *Copy* button copies the trace data 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.

All of the parameters 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). This option is handy for setting loudspeaker EQ curves. The example application on *Setting an Equalizer for a Loudspeaker* in Chapter 6, beginning on page 101 talks about this in more detail.

Copy to ASCII (ASCII Export)

To export a captured trace to ASCII, select the trace that you want to export in the data bar, then click the *More* button and select *Copy To ASCII* in the pop-up menu. Alternately, you can right-click (or [Ctrl]+click on Mac) on the trace that you want to export and select *Copy To ASCII* in the context menu. Either action copies the trace data to the operating system’s clipboard in 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. The ASCII export function uses current display settings to determine the format of the output. For example, if 1/12-octave banding is selected for spectrum data, then the ASCII output will be in 1/12rd octave format. For transfer function traces, smoothing is applied to exported data if applicable. Note that there is no option for saving ASCII data directly to a text or .csv file, however it is a simple matter to paste data from Smart into any text editor such as Notepad (Windows) or TextEdit (Mac) and save text files from there.

Import ASCII

In addition to its own reference file formats, Smart 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 an ASCII text file, click the *More* button at the bottom of the data bar and select *Import ASCII* from the pop-up menu. This opens the *ASCII Import* dialog window. You will be creating a regular stored data trace so you need to specify an *FFT Size* and *Sample Rate* – together these two things determine frequency spacing. The current active measurement type determines the *Trace Type*.

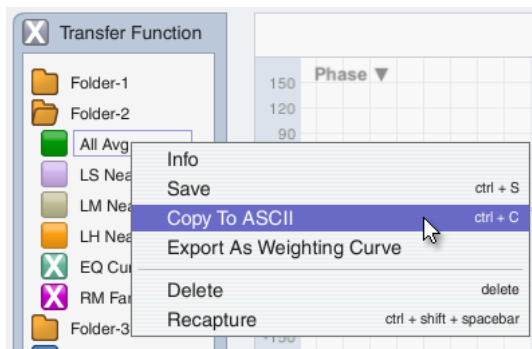


Figure 24: The right-click ([Ctrl] + click on Mac) context menu for stored data traces.

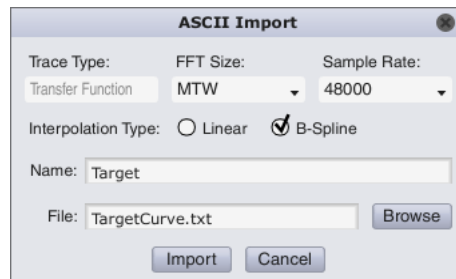


Figure 25: ASCII Import dialog for importing trace data from text files.

The frequency spacing of data in the text file does not need to match the precise FFT bin frequencies for the *FFT size* and *Sample Rate* that you select. Smaart will automatically interpolate the frequency points that it needs from whatever set of coordinates you supply, using *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 sets the name of the trace to be created.

Assuming that you do not want to type the full path and file name of the file that you wish to import, click the *Browse* button to bring up the Load File dialog, to 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 on page 176 for more details).

Once all parameters are set, click the *Import* button to import the data from the file. The ASCII Import dialog will close and your new imported trace should appear in the data bar.

Weighting Curves

Weighting curves are used in Smaart to shape the display of live or stored measurement data, giving more or less weight to some frequencies relative to others. Another way to put it might be to say that filtering in the time domain becomes 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.

There are two ways to add a weighting curve to live measurements in Smaart. You can open the *Measurement Config* dialog (*Options* menu > *Measurement Config*) and use the *Weighting* selector in the global settings section which applies the curve to all measurements in the currently selected group that subscribe to the global setting. Alternatively, you can navigate to the measurement settings tab for an individual spectrum or transfer function measurement in *Measurement Config*, un-check the *Use Global* check box next to the *Weighting* field, and then select a weighting curve that applies only to that measurement.

To add a weighting curve to a captured data trace, right-click (or [Ctrl]+click on Mac) on the name of the trace in the data bar, select *Info* from the context menu, then use the weighting selector in the *Trace Info* dialog window. There is no global weighting option for stored data traces.

Smaart comes with built-in curves for standard A, B and C weighting functions used for SPL and LEQ measurements and the X curve used in cinema systems, with normal and inverted versions of each. In addition to the built-in curves, you can also add user-defined curves, by either importing data from text files or exporting a captured transfer function traces as a weighting curves. 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.

Export Stored Data Trace as Weighting Curve

To export a stored transfer function measurement trace as a weighting curve, make sure the active measurement type is *Transfer Function*, then select the trace that you want to export in the data bar, click the *More* button at the bottom of the select *Export as Weighting* curve from the pop-up menu. A small dialog window pops up where you can specify a name for the new weighting curve and when you click *OK*, the new curve is added to the weighting curve menus for live measurements and stored data traces. Smaart automatically creates an inverted version of the curve as well and appends the notation “*Inv*” to its name.

Import Weighting Curve

To import a weighting curve stored in an ASCII text file, select *Import Weighting Curve* from the *File* menu to open the Custom Weighting Curves dialog window, then click the *Import* button. This opens the Import Weighting Curve window where you can navigate to the file containing your custom weighting curve and open it to import the curve data.

Delete Weighting Curve

To delete a custom weighting curve, select *Import Weighting Curve* from the *File* menu to open the *Custom Weighting Curves* dialog window, then click the name of the curve that you want to delete and click the *Delete* button. Deleting a weighting curve removes both the normal and inverted versions. Note that the A, B, C and X 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 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’re done.

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 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, use the on-graph trace menu to bring it to the front from or press the [Z] key on your keyboard repeatedly until it comes up, then press [Alt/Option] + [Q] or select *Capture Quick Compare* from the *Options* menu to copy the trace into memory.

When you have captured the reference curve, press the [Q] key on your keyboard or select *Toggle Quick Compare* from the *Options* 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 *Options* menu again. To replace the reference, simply repeat the *Capture Quick Compare* procedure.

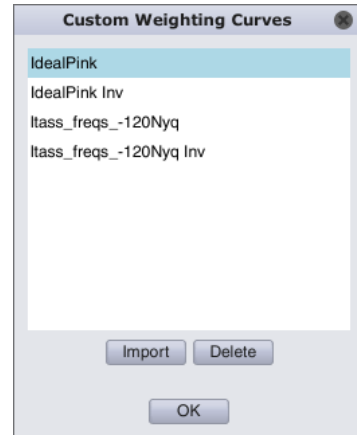


Figure 26: Custom Weighting Curves dialog, accessible by selecting *Import Weighting Curve* from the *File* menu.

Target Curves

A Target Curve in Smart is a line with a specified shape, drawn on banded RTA displays (only) at a specified level. Target Curves cannot be grabbed or moved or averaged with other traces as live and static data traces can and they do not appear on unbanded (narrowband) spectrum displays. Their position is fixed with the exception of moving up or down with banding changes. The cursor readout ignores the Target Curve.

The purpose of the Target Curves feature, as the name might imply, is just to be visible on the screen as a reference or target response curve, for example when measuring background noise or tuning a system to a target response curve specified in fractional octave form such as the cinematic X curve or a standardized noise-masking curve.

To display all target curves that are currently selected for display, select *Show Target Curves* from the *Options* menu or press [X] on your keyboard. Pressing [X] again or reselecting *Show Target Curves* from the *Options* menu hides all target curves when visible.

To view and manage all available target curves, select *Target Curves* from the *Options* menu or press [Alt/Option] + [X] on your keyboard. This opens the *Target Curves* dialog window. Here 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.

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. 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.

Importing Target Curves

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]
```

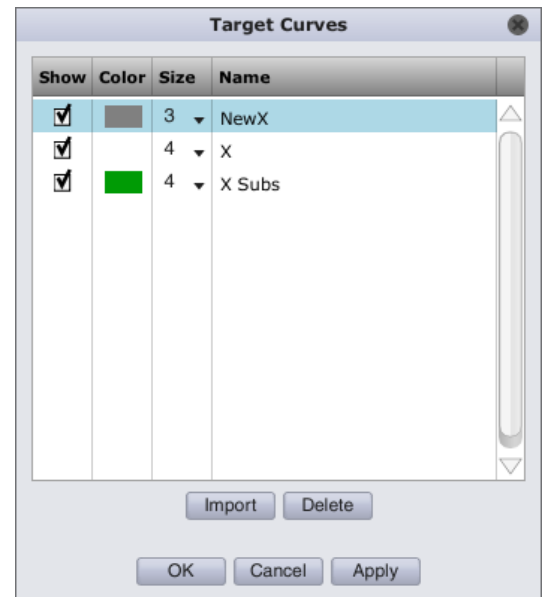


Figure 27: The Target Curves dialog window.

Chapter 2: Finding Your Way Around in Smaart

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, and so you may want to just set the color for a new trace to FF808080 initially for a neutral gray color, and then use the color 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 the 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.

The following example should produce an updated cinematic X curve (per SMPTE ST 202:2010), if you type the text into a text file and then import the file as a target curve. Note that the frequency and magnitude values below the header text are separated by [Tab] characters.

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

10      65.1
12.5    66.1
16      67.1
20      68.1
25      69.1
31.5    70.1
40      71.1
50      72.1
2000    72.1
2500    71.1
3150    70.1
4000    69.1
5000    68.1
6300    67.1
8000    66.1
10000   65.1
12500   63.1
16000   61.1
20000   59.1
```

In the example above, magnitude values are referenced to sound pressure level (SPL). This means that the input driving an RTA measurement in Smaart would need to be calibrated to SPL (see page 77) and the *Plot Calibrated Levels* option must be enabled in Spectrum options (see page 63) to display live measurement data in proper in relation to the target curve. The curve is specified in 1/3-octave intervals (*Band: 3*) and because the X curve is a flat line between 50 and 2000 Hz, values for the 1/3-octave bands from 63 Hz to 1.6 kHz have been omitted. Smaart needs only the two end points to draw the line.

Signal Generator

Smaart's signal generator is a versatile and highly configurable signal source capable of driving all types of frequency and time domain measurements. It is capable of generating 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 stored in .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 when we get 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 controls strip in the upper right of the main Smaart window in both real-time and impulse response modes. These top-level controls show you the basic signal type and output level, and provide a convenient way to turn the generator on and off and adjust the output levels. The On/Off button on the left displays the basic signal type that is currently selected (*Pink Noise*, *Pink Sweep*, *Sine*, *Dual Sine* or *File*). The output level readout in the center doubles as a button to open the *Generator* option dialog, which is also accessible by selecting *Signal Generator* from the *Options* menu or by pressing [Alt/Option] + [N] on your keyboard. The plus and minus buttons increase or decrease output level.

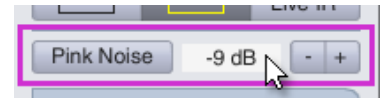


Figure 28: On-screen signal generator controls.

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 are monaural and when two output channels are selected, the same signal is sent to both. If a stereo .wav or .aiff file is selected for file based signal generation, only the left channel is used and the right channel is ignored.

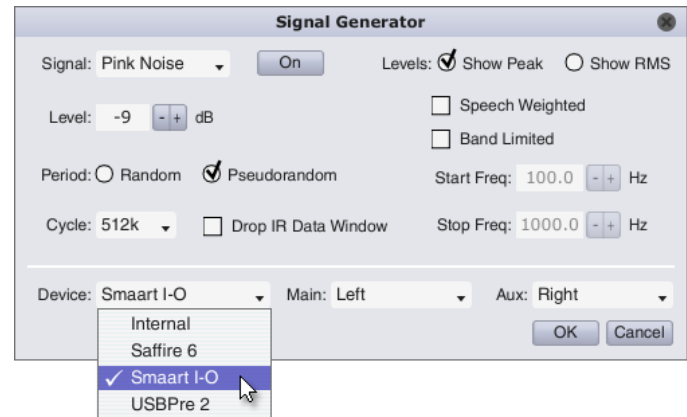


Figure 29: Signal Generator output device and channel selection.

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, however it is randomly seeded 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.

For real-time measurements in general, the maximum setting of 512K is generally a good choice for cycle. That works out to about 11 seconds at 48 kHz sampling rate, which is long enough to make repeats unnoticeable but still short enough for averaged measurements to settle relatively quickly. 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. When *Show Peak* is selected, the signal level shown in the *Level* field shows the peak signal level. When *Show RMS* is selected, Smaart subtracts the 12 dB.

Pink Sweep

The *Pink Sweep* signal is a logarithmic sinusoidal sweep primarily intended for use in impulse response measurements for room acoustics analysis.

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 typically use little or no averaging when using sweeps to make an IR measurement, as there is little to be gained by averaging multiple measurements when using sweeps.

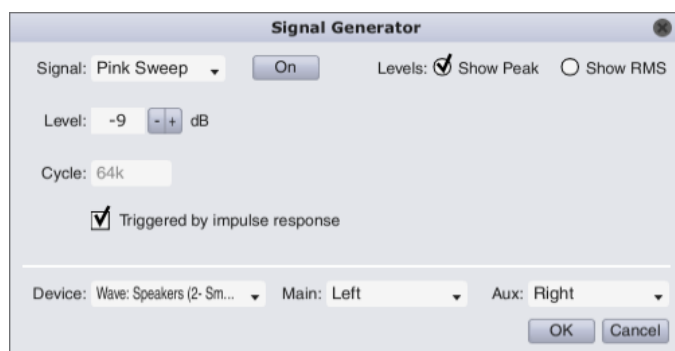


Figure 30: Signal generator options for Pink Sweep (log-swept sine) signals.

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 31. 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.

Note that Smart 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). When *Show RMS* is selected, Smart simply subtracts 3 dB from the peak level.

File-based Signals

In addition to internally generated signals, Smart 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, Smart will scale the signal to a peak level of 0 dB normalized full scale.

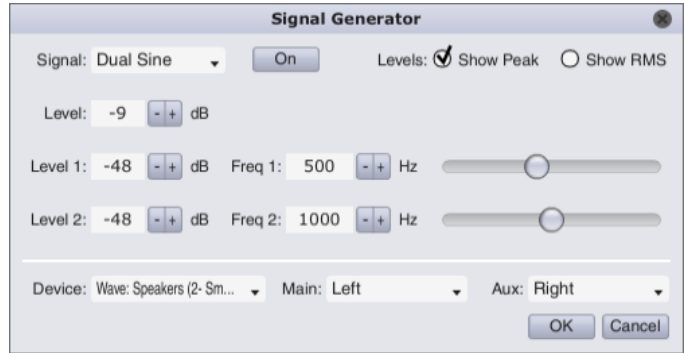


Figure 31: Signal generator options for Dual sinewave signals.

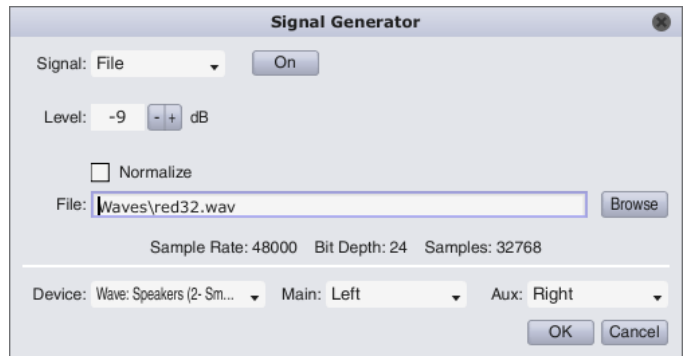


Figure 32: Signal generator options for file based signals.

Chapter 3: Configuring Smaart for Real-Time Measurements

The first time you run Smaart, the program opens to an RTA graph and immediately asks if you want to set up a new Spectrum measurement. If you switch the display to a transfer function (magnitude or phase graph), Smaart will prompt you again to set up a transfer function measurement. You will need to set up at least one measurement of either type before you can make any live measurements. Assuming that you are running the full version of the program, Smaart will remember the measurements you set up the next time you run the program, so if you always use the same input device(s), you may only need to do this once. (The demo version of Smaart does not remember its settings when you shut down.)

Live measurements are configured from the aptly named *Measurement Config* window, accessible by clicking the button with the hammer and wrench icon next to the *Group* selector on the control strip (visible in the upper right corner of Figure 33) or by selecting *Measurement Config* from the *Options* menu. A live measurement in Smaart consists primarily of one or two input device and channel assignments, the display color for graphing – Smaart assigns trace colors automatically to new measurements but you can change the colors if you like. Internal processing and display parameters such as FFT size, averaging and smoothing (as applicable) can be set specifically for each measurement but are more typically set globally, for all measurements of the same type, meaning that your main concern when setting up a new measurement is usually just selecting the input channel(s) to drive it.

The default live measurement settings in Smaart 7 have been carefully selected to work well for the large majority of typical applications and really should not require much adjustment for most jobs. The real-time mode control strip includes on-screen controls for averaging, banding (RTA) and smoothing (transfer function) because those are the settings that tend to be changed most frequently. We encourage newer users in particular to stick close to the default settings for other measurement parameters and change them only with good reason.

Configuring Live Measurements

Each live measurement in Smaart is essentially a complete real-time spectrum or transfer function analyzer. These can be driven by any audio stream(s) that your computer can access. There is no set limit on how many measurements that you can create and run simultaneously. There may be a practical limit, depending on factors such as available computing resources (RAM, CPU and GPU) and how many microphones and input channels you can afford, but Smaart imposes no limits of its own.

Live measurements are organized into groups and one group of each type (Spectrum and Transfer Function) is always active. Controls for each measurement in the active group appear in the lower portion of the control strip when a graph of the corresponding type is the active display.

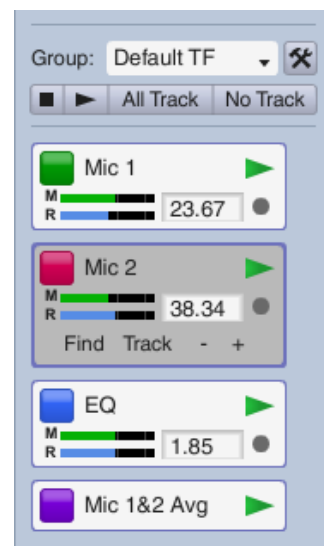


Figure 33: Live transfer function measurement controls on the real-time mode control strip.

If you have a relatively simple hardware setup and work with the same device(s) all of the time, you can almost ignore the group feature and just use a single group of each type for everything you do. However if you have a more complex measurement setup or work with several different hardware devices in different scenarios, it may be worth your time to set up multiple measurement groups to streamline switching between tasks and/or hardware configurations.

Creating and Managing Live Measurements

Live measurements and groups are created, edited, and deleted through the *Measurement Config* control panel. To access *Measurement Config*, you can click the little hammer and wrench button next to the *Group* selector in the control strip (see Figure 33), or select *Measurement Config* from the *Options* menu, or press [Alt/Option] + [G] on your keyboard. Double-clicking the control set for any existing live measurement on the control strip in real-time mode will also take you directly to its measurement details tab in *Measurement Config*.

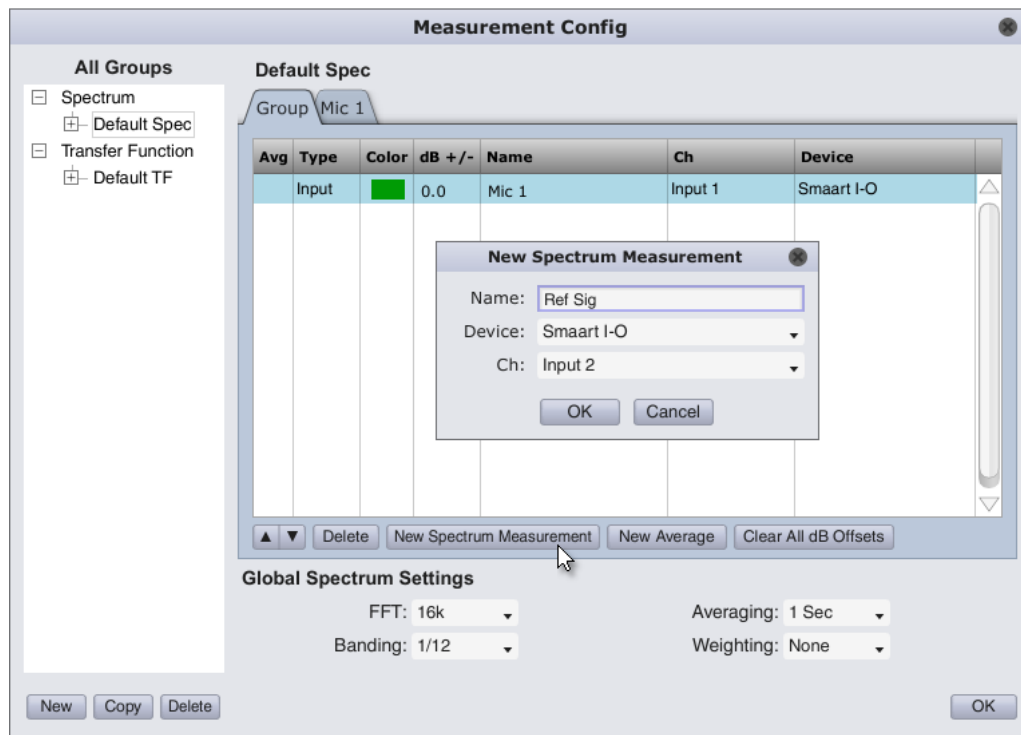


Figure 34: Creating a new live spectrum measurement in *Measurement Config*.

Measurement Groups

On the left side of the *Measurement Config* window is a “tree” control – a hierarchical list with expandable branches – labeled *All Groups*. The tree view lists all of your configured measurement groups and the individual measurement objects of each type. The little +/- buttons to the left of the measurement type and group names expand and collapse the branches. The selection in the tree view determines what you see on the right side of the *Measurement Config* window. Selecting the name of a group brings up its *Group* tab on the right. If you expand a group and click on one of the measurements that it contains, the measurement settings tab page for the selected measurement appears on the right.

Below the tree control are three buttons labeled *New*, *Copy* and *Delete*. The *New* button creates a new group of the currently selected type (Spectrum or Transfer Function). The *Copy* button can be used to create a new copy of the currently selected group. You will be prompted for a name for the new group in either case and then your new group will be appended to the tree. The *Delete* button deletes the currently selected group. Note that if you delete the last remaining group of either type, Smaart immediately creates a new empty group of the same type called *Default Spec* (spectrum) or *Default TF* (transfer function).

Creating a new Live Measurement (Spectrum or Transfer Function)

To create a new spectrum (single-channel) measurement, make sure that a *Spectrum* group is selected in the tree view in *Measurement Config* and that the *Group* tab is selected on the right, and then click the *New Spectrum Measurement* button located below the measurements table. A pop-up window opens, prompting you to specify a name for the new measurement and select an input device and channel to drive it. Note that the devices and channels are listed by their friendly names where applicable (see *Managing Audio Input and Output Devices*, beginning on page 50). When you click *OK* to exit the pop-up window your new measurement appears in the measurement table on the *Group* tab and new measurement settings tab is added to the group with additional options and settings for that measurement.

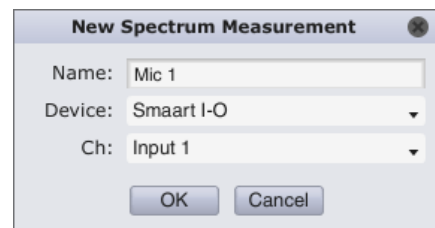


Figure 35: The New Spectrum Measurement dialog.

In many, if not most cases, this is actually all that you need to do to create a new measurement. Smaart assigns a unique color to each new measurement automatically and the default measurement settings tend to work quite well for most applications, so there is typically little need to adjust anything else.

To create additional measurements, click the *New Spectrum Measurement* button again and repeat the procedure. As a rule, you will want to set up a spectrum measurement for every input channel that you plan on using. Even if you will be focusing primarily on transfer function measurements, the RTA view of each input is still extremely useful for checking signal levels and for troubleshooting purposes.

Creating a new transfer function measurement works much the same way as a spectrum measurement except that you need to select two input channels. First, make sure that a *Transfer Function* group is selected in the tree view on the left and the *Group* tab is visible on the right, and then click the *New TF Measurement* button. A dialog window opens prompting you to enter a name for the new measurement and select an input device and two input channels.

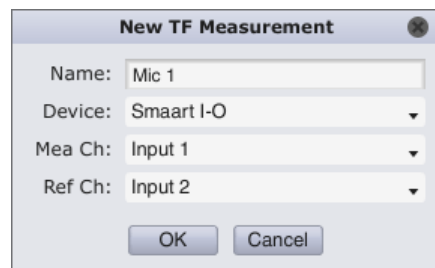


Figure 36: The New TF (transfer function) Measurement dialog.

The measurement signal channel (abbreviated *Mea Ch*) will typically be a microphone input or perhaps the output of a device under test if you are making an electronic measurement. The reference signal channel (*Ref Ch*) should be an input to which you are sending the same signal being used to drive the device or system under test.

Notice that there are quite a few more measurement settings for transfer function measurements than there are for spectrum measurements, but here again, selecting input channels to drive your new live measurement may be all that you need to worry about in most cases. Smart handles color selection automatically and the first-run defaults for all the other settings are carefully chosen to work well for most tasks. There may be good reasons to tweak measurement settings for specialized applications, but before you change anything make sure you know the reason why you are doing it.

Live Averages

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

To create a new live average, click the *New Average* button below the measurements table on the *Groups* tab in *Measurement Config*. You will be prompted to name the new averaged measurement regardless of type. If you are creating a new spectrum average, you will also be asked to select *Power* or *Decibel* averaging. For transfer function live averages, you have the option of using coherence weighing. (If you change your mind later in either case, just go to the measurement settings tab for the measurement and change your selection.) For more information on power versus decibel averaging and coherence weighing, please refer to the topic on *Spatial Averaging* on page 16.

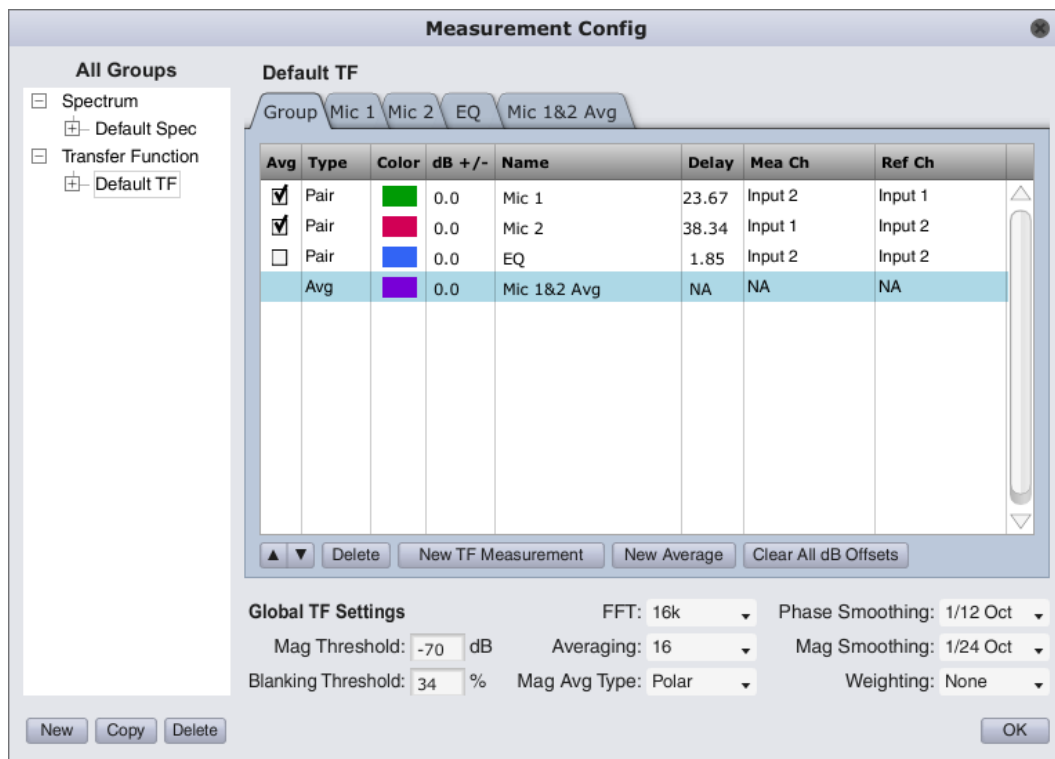


Figure 37: Selecting measurements for a live average.

As with new live measurements, when you click *OK* to create the new live average, its name is added to the measurements table and a new settings tab for the measurement is added to the group. Notice however that when a live average is selected in the measurements table, a square box appears next to

the name of each live measurement in the *Avg* column of the table. You can click these buttons with your mouse to select the measurements that you want to include in your averaged measurement.

Measurement Group Table

The measurement group table in the *Group* tab in *Measurement Config* gives you an overview of all the measurements in a group. As we discussed in the previous topic it provides the mechanism for selecting source measurements for live averages and some of the other properties of each measurement are directly editable from this tab as well.

The *Color* column shows you the display color for each measurement in the group. If you want to change any of these, you can click on any of the color tiles to bring up a color picker dialog.

The values in the *Name* and *dB +/-* column are also directly editable. You can click any entry in these with columns with your mouse to make it editable, type in the new name or value, then press the [Enter] key on your keyboard to set the change.

Applying a dB +/- offset moves the selected trace up or down on applicable charts in chart area of the main window. Positive dB +/- values move the trace up on the chart by the specified number of decibels. Negative numbers move it down. If the measurement that you are editing is visible on a graph in the main window, you can see the trace move up or down when you press the [Enter] key to set the change. Note that dB offsets are not stored in Smart's config file, meaning that they get zeroed out when you close the program.

Transfer function measurements have one additional column of editable settings and that is the *Delay* column. To change the reference signal delay for any dual-channel measurement in the group, click on the value that you want to change, type in a new value, then press the [Enter] key to set the change.

Notice that the measurement control blocks on the control strip in the main window are arranged in the same order as they appear in the table on the *Group* tab in *Measurement Config*. If you want to rearrange the list, click on the name of a measurement that you want to move and use the up/down arrow buttons (▲ | ▼) below the table to move it up or down in the list.

Global and Local Measurement Parameters

Below the *Group* and measurement settings tabs in each group is a section labeled *Global [Spectrum or TF] Settings*. Some of these simply apply to all measurements in the group, others can be localized for individual measurements. If you switch to one of the individual measurement settings tabs, you will see that some settings fields have a *Use Global* check box next to them that is selected by default. When this box is checked, changes to the global settings for the selected are automatically applied to the measurement. Unchecking the *Use Global* check box for any of these parameters activates the local control so that global changes will not affect it.

Global Settings for Spectrum Measurements

- *FFT size* (in samples) determines the time constant and frequency resolution of the frequency domain displays. This setting applies to all spectrum measurements.

- *Banding* sets the frequency resolution for RTA and Spectrograph plots. Options include *None* (narrowband frequency resolution), octave banding (*Oct*) and fractional octave banding from *1/3rd* to *1/48th* octave.
- *Averaging* specifies the length of time over which individual measurements are averaged to smooth and stabilize the traces or bar graphs on the screen. Please refer to the topic on *Averaging Over Time (Temporal Averaging)*, beginning on page 15 for details on averaging options. Averaging can be set globally or locally for each measurement.
- *Weighting* applies a weighting curve to spectrum 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. For more information on weighting curves, see *Weighting Curves* on page 37.

Global Settings for Transfer Function Measurements

- *Magnitude Threshold* sets the dB FS value that the reference signal must exceed before new data is accepted into the measurement at a given frequency. This setting applies to all transfer function measurements.
- *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 all transfer function measurements.
- *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. Averaging helps to smooth and stabilize the traces or bar graphs on the screen. Please refer to the topic on *Averaging Over Time (Temporal Averaging)*, beginning on page 15 for details on averaging options. Averaging can be set globally or locally for each measurement.
- *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. This setting applies to all transfer function measurements.
- *Phase Smoothing* sets the amount of smoothing used for the Phase display. This option can be set globally or locally for each measurement.
- *Mag Smoothing* sets the degree of smoothing type for magnitude traces. This option can be set globally or locally for each measurement.
- *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. For more information on weighting curves, see *Weighting Curves* on page 37.

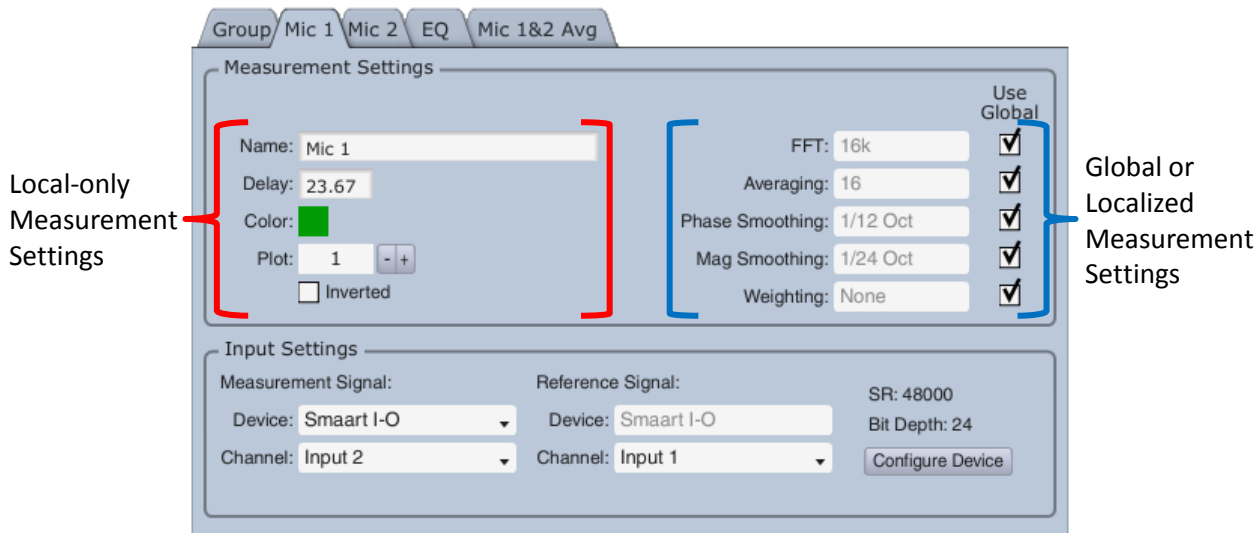


Figure 38: Local and Global measurement settings for a transfer function measurement.

Local (only) Measurement Settings

Some local measurement settings do not have global equivalents. Local-only settings vary somewhat depending on the measurement type but three that apply to all live measurements are *Name*, *Color* and *Plot* assignment. Measurement name and display color can be edited from the *Group* tab or on the individual measurement tabs. Clicking the *Color* tile in either location brings up a color-picker dialog window, where you can select a new display color for the measurement.

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

Local-only settings for live transfer function measurements include a *Delay* field, which sets the reference signal delay for the measurement and an *Inverted* check box. When the *Inverted* box is checked, Smart will display the measurement upside-down. This can be a handy option when setting loudspeaker EQ curves – a cut filter will plot as a boost and a boost filter as a cut when the measurement is inverted, making it easy to create a complimentary EQ curve. The example application on *Setting an Equalizer for a Loudspeaker* on page 101 talks about this in more detail.

Transfer function live averages also have a *Coherence Weighted* check box that turns coherence weighting on and off. *Spectrum* live averages have a pair of radio buttons for selecting *dB* or *Power* averaging. We talked about these options in Chapter 1, in the topic on *Spatial Averaging* on page 16.

Managing Audio Input and Output Devices

Smart can interface with multiple audio input devices simultaneously and can stream output signals to any output device connected to your computer as a stimulus source for measurements. Smart can

acquire input signals from devices with Wave (native Windows audio, also commonly called WDM), Mac CoreAudio, or ASIO drivers that are recognizable by the computer’s Operating System (OS).

Smaart can work with signals from multiple Wave or CoreAudio sources simultaneously. On Windows systems, you can use only one ASIO device at any one time, but other Wave audio devices may be used in conjunction with an ASIO device.

When Smaart v7 starts up, it checks to see what I/O devices are available to your computer and builds its list of input devices. Once a device is discovered, it will continue to show up in your available device lists in subsequent runs even if it is not connected, unless you delete it from your configuration. This enables you to run Smaart without disturbing existing measurement setups when your computer is in one place and one or more of your audio I-O devices are elsewhere.

If you work with a number of different audio devices, particularly devices with a lot of input channels, this can result in lengthy lists of devices and channels and so Smaart lets you choose which devices you want to see and ignore the ones that you don’t use. Additionally, you can assign your own Friendly Name to any available audio device and/or any input or output channel. For input channels, you can also specify a calibration offset for sound level measurements (SPL/LEQ) and a correction file for a specific microphone connected to the channel if you like.

Audio Device Options

To see the list of available audio devices, click on the *Options* menu and select *Audio Device Options*. This will pop up a dialog window of the same name, as shown in Figure 39. In the upper portion of this window, you can set the *Sample Rate* and sample word size (*Bits per Sample*). These are global settings that apply to all input and output devices. The lower section has two tabs, one for input devices and another for outputs – these are considered to be separate virtual “devices” even when they reside on the same physical device.

On each tab is a table with four columns:

Ignore – Checking the *Ignore* box for a device removes that device and all its channels from available device lists elsewhere in Smaart. Clicking the box for an ignored device clears the check mark and restores it to applicable device lists.

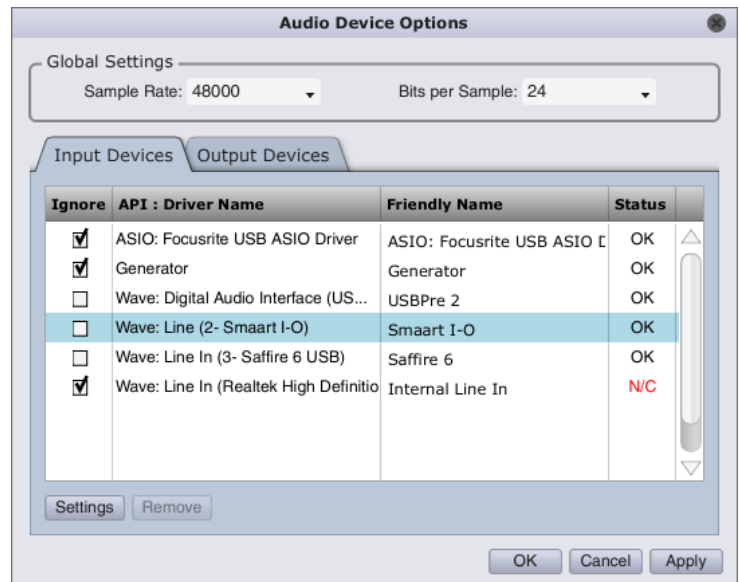


Figure 39: The Audio Device Option dialog.

The *Ignore* (audio device) feature has proven to be a frequent source of confusion in Smaart 7. When using this feature, please note that your selections in the *Ignore* column designate devices that you **don’t** want to see in device lists, not the devices that will remain visible.

API : Driver name – API stands for Application Programming Interface. An API is a defined interface that enables two or more software processes (e.g., applications, operating systems and device drivers) to talk to each other and exchange data.

On Mac OS X, all devices that you see will be CoreAudio. On Windows machines, you may see both ASIO and Wave API devices. The name after the colon is the device name reported by the driver. Note that many multichannel audio devices for Windows come with both ASIO and Wave API drivers. Since the Wave API in Windows supports only stereo recording (and multichannel cinema style output), multichannel input devices are represented as several virtual devices with two input channels each. It is a good idea to click ignore for all the Wave “devices” in that case, both to clean up device lists and also to prevent you from inadvertently trying to access the same device via two different APIs at the same time.

Friendly Name – The friendly name for a device can be any name that you want to assign it. These are the names that you will see when configuring measurements in Smaart. Entries in the *Friendly Name* column are directly editable. Just click on the one that you want to change and type in whatever you want it to be. As with most text entry fields in Smaart, remember to hit the Enter key after editing to apply your changes.

Status – The status for a device can be “OK,” meaning that Smaart was able to connect to it successfully at start-up time, or NC (not connected). NC can mean *really* not connected or, if the device is present and connected, it could indicate some hardware or software problem that prevented Smaart from successfully communicating with it on start-up – this could be due to a hung device driver, a loose cable, or perhaps the device has become unresponsive and needs a reboot. In that case, remember that Smaart scans for audio devices *only* on start-up, so 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 NC device by selecting it in the list and clicking the *Remove* button.

Configuring Input and Output Channels

To set up friendly names and other options for individual input channels, select the device that you want to configure on the Inputs tab, and then click the Settings button below the device table. This will pop up another dialog window (see Figure 40) with a table of input channels for the device.

Channel Name – The first column in the table is the official (driver-reported) channel names. These are not editable.

Friendly Name – The friendly name for a channel can be anything you want it to be. For example, if you had a 4-channel input device you might name your inputs, Mic 1, Mic 2, Mic 3 and Ref Sig. These names would then be the channel names you would see when setting up measurements. Entries in the *Friendly Name* column are directly editable. Just click on one that you want to change and type in whatever you want it to be. As with most text entry fields in Smaart, remember to hit the [Enter] key after editing to apply your changes.

Calibration offset – The numbers in the *Cal. Offset* column represent the difference in decibels between each input channel’s full-scale level and its calibrated level. A calibration offset of zero means the input is calibrated to digital full scale. When an input is calibrated to 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

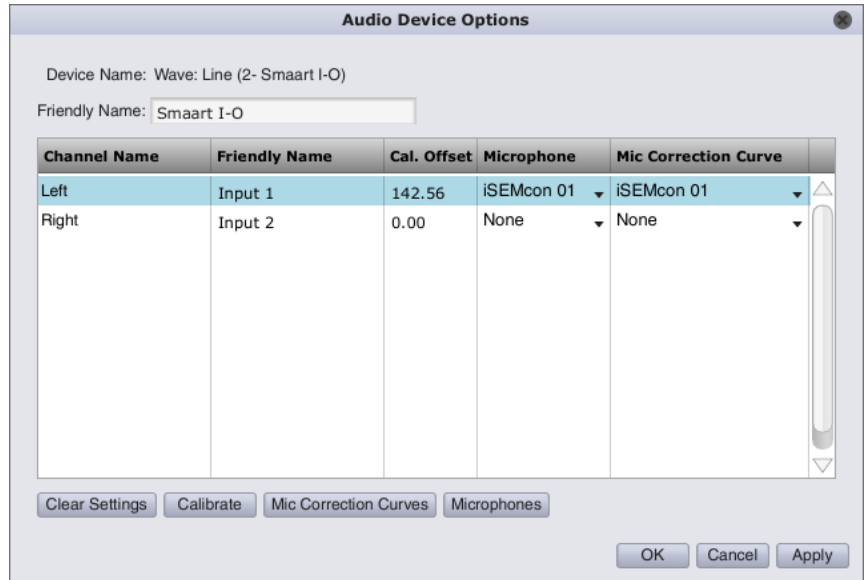


Figure 40: Configuring an audio input device in Smaart.

in. We will talk more about calibration when we get to sound level measurements. For now, just note that this is one of the places where that number can be found.

Microphone – If you have a Smaart I-O, another way of calibrating for sound level measurement is to tell Smaart the sensitivity of your microphone and let it calculate the required offset for you. To create a named microphone, click the *Microphones* button below the channel table, then click the *Add* button in the pop-up window. Give your new microphone a name and enter its sensitivity in dB/Pascal, then *OK* your way out of the *Create Microphone* and *Microphones* pop-up windows. Now, when you click on any line in the *Microphone* column of the channels table you should see your new microphone as an available option. If the Smaart I-O is connected and the control applet is running, Smaart will set the channel calibration when you click *OK* to close the device configuration window.

Mic Correction Curve – If you have individually measured response data for your microphone(s) Smaart can use this information to flatten out lumps and bumps in the microphone’s response in spectrum and transfer function measurements. Correction curves can be imported from comma- or tab-delimited ASCII text files with one frequency (in Hertz) and one magnitude value (in dB) per line. To add a new correction curve to your configuration, click the *Mic Correction Curves* button below the channels table. In the pop-up window, click the *Import* button, then navigate to the location of the file containing your correction curve and open it. Click *OK* to exit the pop-up window and click any entry in the *Mic Correction Curve* column of the channels table. You should now see your imported curve in the list. For more information on formatting correction curve files, please refer to *Appendix G: Text File Formats for ASCII Import*, beginning on page 176.

Clear Settings button – To clear all settings for a device, click the *Clear Settings* button below the channels table on the right. This will clear out any calibration offsets, microphone sensitivities and correction curves that you have configured and reset friendly names for the device and all its channels

Chapter 3: Configuring Smart for Real-Time Measurements

to the driver-reported default names. (A warning dialog pops up first in case you click the button by accident.)

Calibrate button – the *Calibrate* button opens the *Amplitude Calibration* dialog with the currently selected input selected channel loaded for calibration. For more information on calibrating input levels in Smart, please see the section on *Sound Level Measurement: SPL* and *LEQ*, beginning on page 75 in Chapter 5.

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 here we have taken a few liberties here for presentation purposes. We have switched from the default low-light color scheme to the higher contrast scheme (*Options* menu > *High Contrast*), there are three live measurements set up and running, and some captured data traces in the data bar.

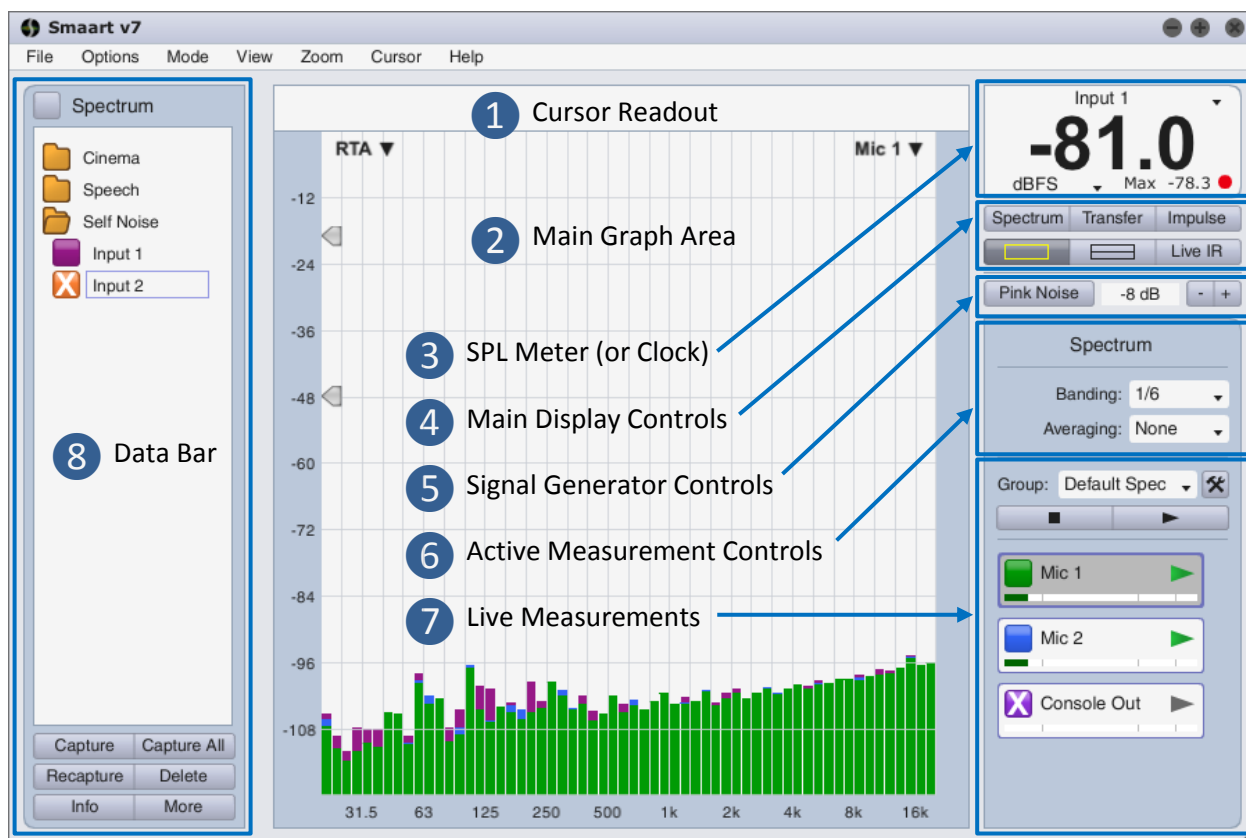


Figure 41: Anatomy of the main window layout for real-time mode.

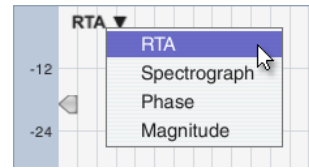
1 Cursor Readout



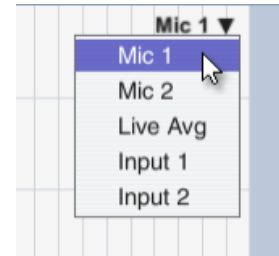
When measurement data is present on a graph, the cursor readout displays numeric coordinates corresponding to the cursor location(s) as you move your mouse over the graphs areas. Numeric coordinates are provided here for the cursor location in units of amplitude/magnitude and frequency or time, as applicable to graph type. The cursor readout works much the same for all graph types in Smaart and is covered in detail in the *Common User Interface Elements* section of the previous chapter, beginning on page 24.

2 Main Graph Area

The main graph area in real-time mode can be divided into one or two graph panes (see #4, Main Display Controls), plus an optional Live IR pane for transfer function displays. 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).



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. If there is a weighting curve applied to the measurement, the weighing curve name appears below the measurement name.



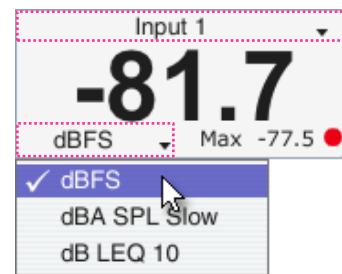
When multiple traces are displayed on the graph, you can click on the label in the upper right corner to display a menu of all visible traces. Clicking the name of any trace in the list will bring it to the front. You can also cycle the z order of a graph by pressing the [Z] key repeatedly.

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.



3 Signal Level / Sound Level Meter (or Clock)

The large numeric readout in the upper right corner of the Smart v.7 real-time mode window 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 digital full-scale. Pressing the [K] key on your keyboard switches the display to a clock.

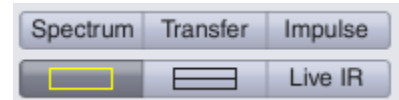


The labels above and below the large numeric value in the center of the display and in the lower left corner function as menus select the input channel driving the display and the type of measurement you want to see. When operating as an SPL or peak meter, you can reset the Max value by clicking the red circle (●) in the lower right corner of the display. In LEQ mode this control changes from red to green (●) once the buffer for the rolling LEQ average is fully populated and clicking it will clear the buffer and restart the LEQ measurement.

Clicking anywhere in center portion of the meter opens the *Sound Level Options* dialog window (also accessible by selecting *SPL/LEQ* from the *Options* menu or pressing [Alt/Option] + [L] on your keyboard). Here you can select weighting curves and integration times independently for both SPL and LEQ measurements, specify a text file for logging, and access the calibration dialog to calibrate your input device(s) and microphone(s) for SPL/LEQ measurement. For more about SPL/LEQ measurements in Smart please refer to the section covering the Signal Level / Sound Level Meter in Chapter 5.

4 Main Display Controls

The first 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 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 several other preconfigured views presets available in the *View* menu, as well as slots for user-defined views. The *Spectrum* and *Transfer* views are given special treatment because they mimic the spectrum and transfer function modes in older versions of Smaart and SmaartLive.
- The *Impulse* button exits real-time mode and switches Smaart into impulse response mode. Impulse response mode (not to be confused with *Live IR*) will be covered in later chapters. Note that in impulse response mode, the *Impulse* button changes to a *Real Time* button that brings you back to real-time mode.
- The two buttons labeled with rectangles divide the main plot area (#2) into one or two graph panes: 1 rectangle, 1 pane; 2 rectangles, 2 panes. The screen clip above shows the single pane option currently selected.
- 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.

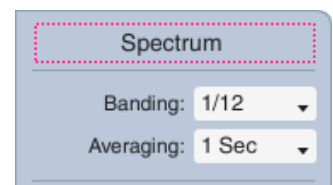
5 Signal Generator

Moving downward, the next group of controls is for the signal generator. The leftmost button in this group displays the currently selected signal type (*Pink Noise* is selected in this example) and turns the generator on and off. The minus and plus sign (-/+) buttons on the right increase/decrease the generator output level. The center area displays the current output level and also functions as a button to open the signal generator control panel (also accessible by selecting *Signal Generator* from the *Options* menu) if you click it with your mouse.



6 Active Measurement Controls

The active measurement control section of the control strip changes according to the graph type displayed in the active graph pane. The label at the top of this section indicates which type of display is (*Spectrum* or *Transfer Function*) currently active. Notice that this 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 display selection.



If the active graph is an *RTA* or *Spectrograph*, the *Spectrum* measurement controls are shown. When a *Magnitude*, *Phase*, or *Live IR* graph is selected, the *Transfer Function* measurement controls appear in this area.

- Active measurement controls for spectrum displays consist of a *Banding* and *Averaging* selector.
- For transfer function measurements, the control set consists of an *Averaging* selector and smoothing controls for phase (*Phase Smooth*) and magnitude data (*Mag Smooth*).

What these controls do depends on the active measurement. The active live measurement is indicated by the background color of its control block in the lower portion of the control strip (see topic # 7 below). This may sound a little complicated but in practice it is very natural to use.

If the current active measurement uses the global setting for averaging, for example, then the *Averaging* selector controls the global setting for averaging, meaning that changes to this setting flow through to all measurements of the same type in the active group that use the global setting. If the current active measurement is not subscribed to the global averaging setting, then the *Averaging* selector affects only the active measurement. The same idea applies to the *Banding* selector for *Spectrum* measurements and the smoothing controls for transfer function measurements. For more specifics on how banding, averaging and smoothing work, please refer to the related topics in the *Fundamental Concepts* section of this guide, beginning on page 13.

7 Live Measurement Controls

The lower portion of the real-time mode controls strip is devoted to live measurement controls. At the top of this section are group level controls. These include the *Group* selector and a button to open the *Measurement Config* dialog window (labeled with a hammer and wrench). Both spectrum and transfer function measurement group have stop all (■) and run all (▶) and buttons below the group selector that turn all measurements in the group off or on. The transfer function measurement group controls also include *All Track* and *No Track* buttons that turn delay tracking on and off for every measurement in the group.



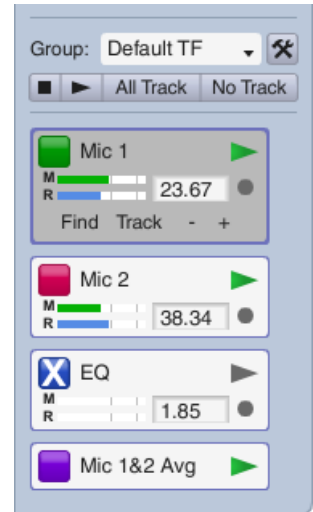
Below the group-level controls are control blocks for each individual measurement in the group – a scroll bar appears if there are too many measurements to display in the space provided so that you can scroll up and down in the list. The controls blocks differ somewhat depending on the measurement type.

Live averages (spectrum or transfer function) have the simplest control set, consisting of just a color tile that matches the display color of the measurement on line and bar charts, the name of the measurement, and a run/stop button (▶). These three elements are common to all types of live measurements. To those, we append 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 (●).

Active Measurement

One live measurement is always designated as the active measurement and will have a different background color than the others. The active measurement is the focus of the active measurement controls (see topic #6) and is the measurement that gets captured when you capture a static “snapshot” of live measurement data. Clicking in the background of a measurement control block makes it the active measurement and brings its data trace to the front on any applicable chart. Conversely, when you select a live measurement on any real-time graph – either by selecting it from the on-graph trace menu or clicking and dragging its data trace with your mouse – it becomes the active measurement.

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 *-/+*. The *Find* button starts the *Delay Finder* for the selected measurement, an automated routine for finding delay times for signal alignment which we will cover in Chapter 6. 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 for the measurement by one sample (about 21 microseconds per click at 48k sample rate, for example).



Run/Stop

Clicking the run button (▶) for a live measurement starts the measurement, clicking again 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 measurements must also be running in order to see any data on the graphs.

Show/Hide

Clicking on the color tile for a live measurement hides its output if it is visible, or un-hides it if it is hidden. When a measurement is hidden, a large “X” appears in the color tile and its data is removed from applicable charts in the graph area.

Notice that when you stop a live single-channel or dual-channel measurement, Smaart automatically hides it. If you unhide a stopped measurement, you will see the last that it data acquired on applicable graphs. This can be a handy way of “freezing” a trace for closer inspection without capturing a stored data trace.

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 Smart:

- 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 window (*Options* menu > *Spectrum*), which we will be talking about later in this section.

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 it 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 Smart.

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.

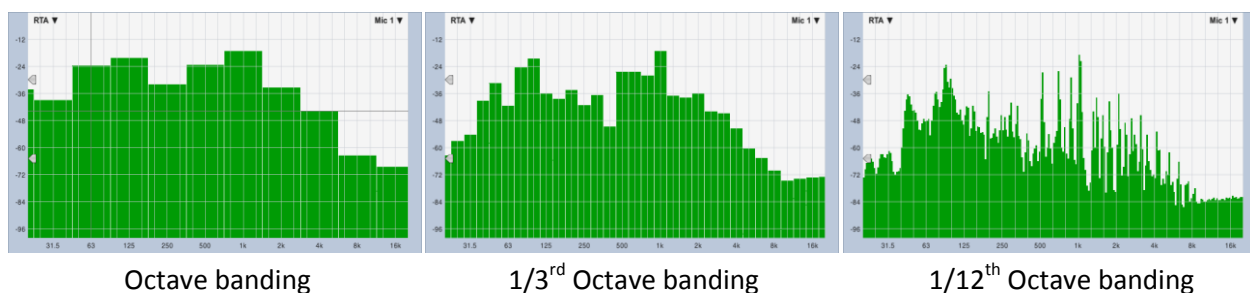


Figure 42: RTA display with Octave, 1/3-octave and 1/12-octave banding.

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.

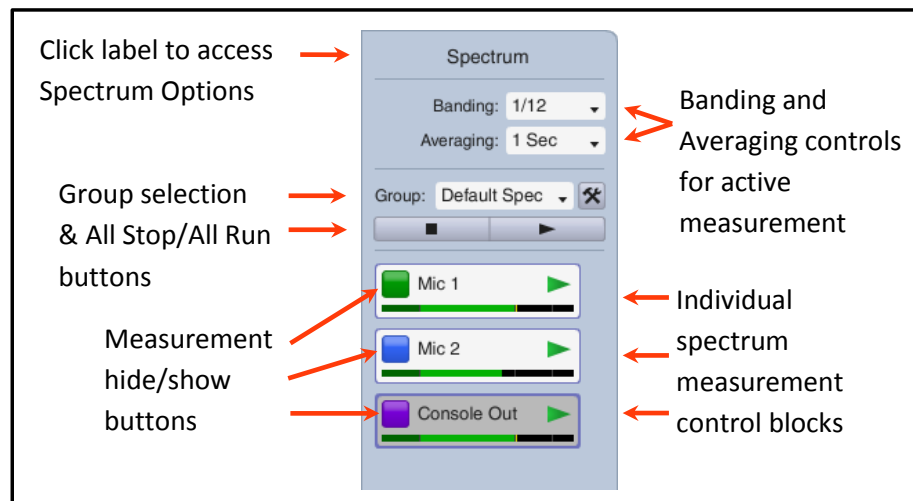


Figure 43: Control Strip for Spectrum measurements.

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 strip on the right side of the main Smart 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 strip 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 Smart 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*).

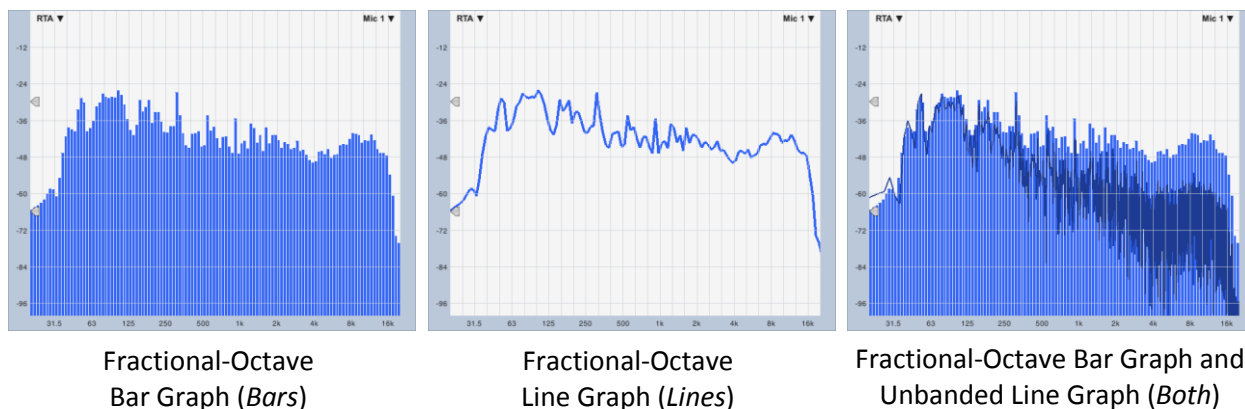


Figure 44: RTA graph types (*Bars*, *Lines* or “*Both*”).

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 on *Peak Hold* in in the *RTA Display Settings* section of the *Spectrum* options (*Options* menu > *Spectrum*). 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 preserved in the captured measurement.

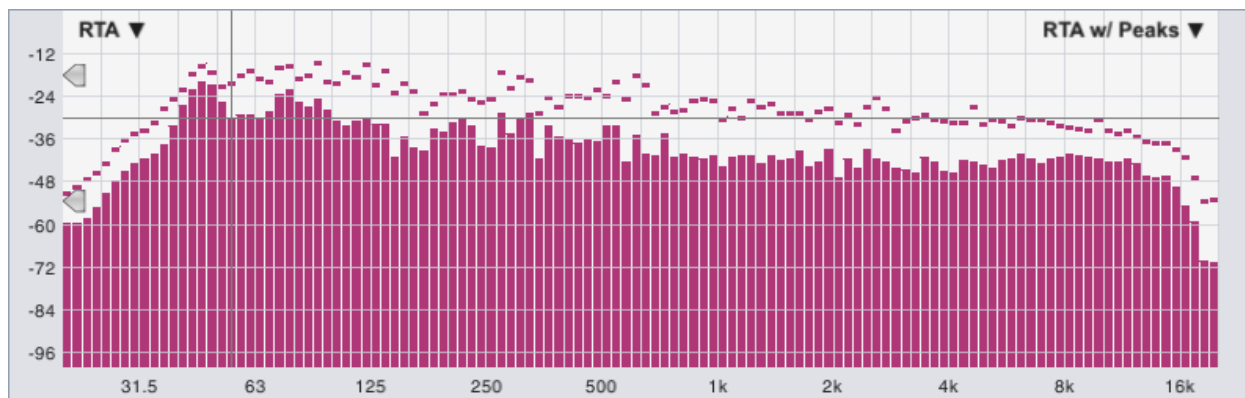


Figure 45: RTA bar graph with peak hold.

Options for *Peak Hold* include *None*, *Timed* or *Infinite*. The default setting is *None*, which turns the feature off. *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, so 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 *Audio Device Options*) 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 measurement 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 *Audio Device Options* from the *Options* menu. Double-click the name of the device that is driving an uncalibrated measurement on the *Input Devices* tab of the *Audio Device Options* dialog to open its settings dialog. You will see that uncalibrated inputs have a *Cal. Offset* 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

Smart'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 46a.

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 46b. 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 46b.

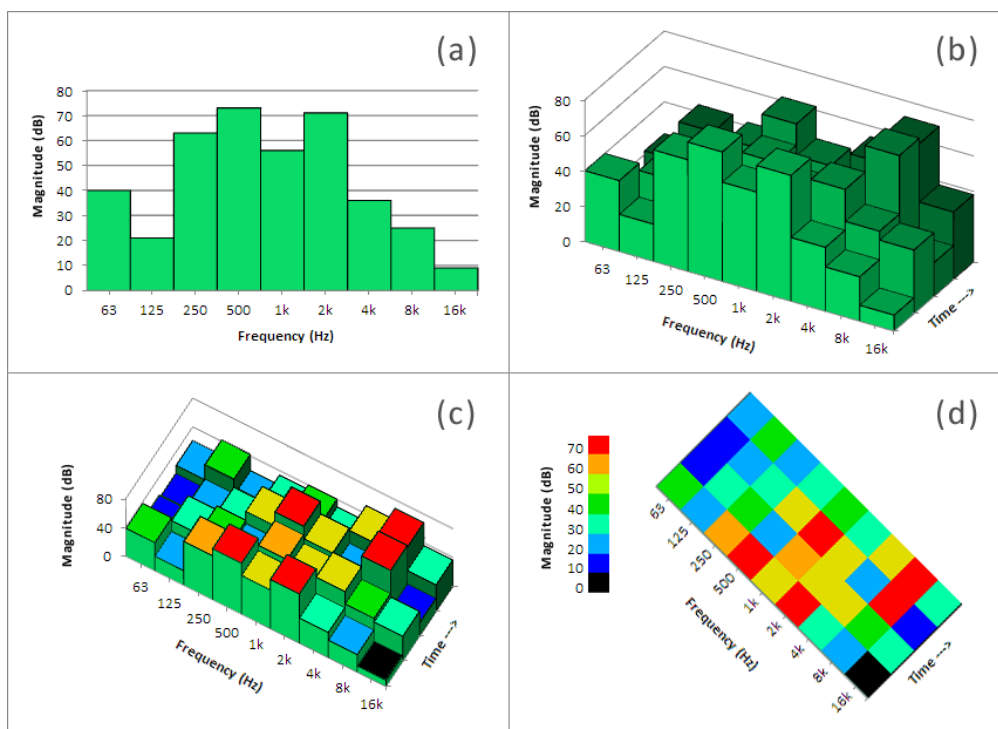


Figure 46: Turning a spectrum analyzer into a spectrograph.

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 46c. 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 46d). 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.

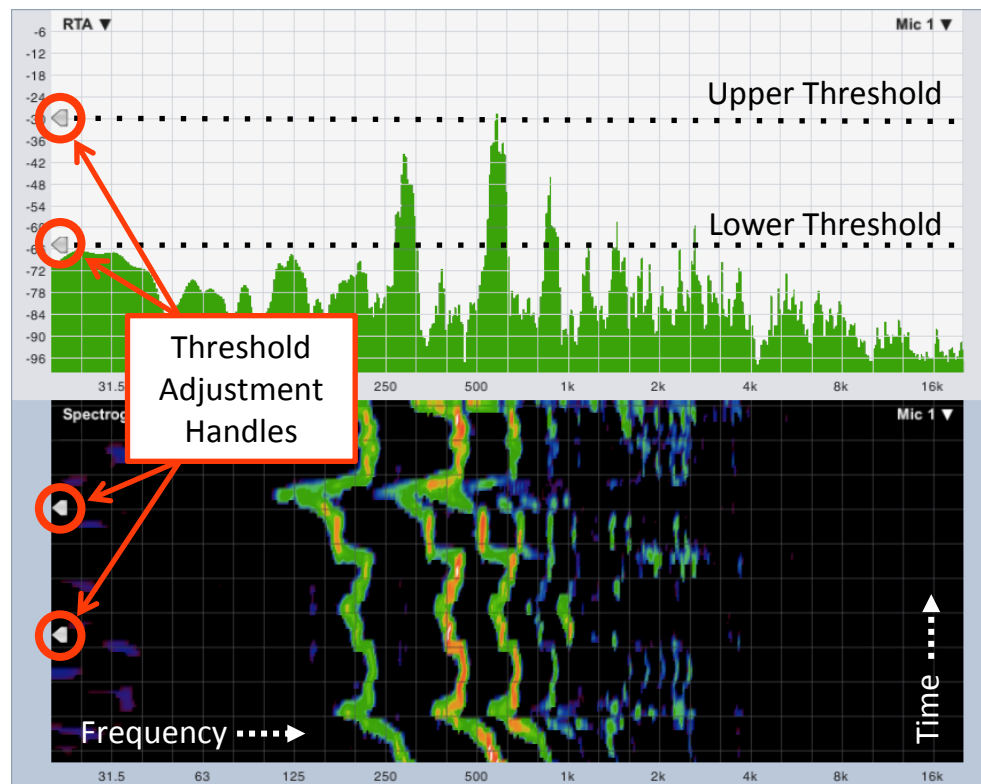


Figure 47: The real-time mode Spectrograph display.

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 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*).

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 Smart 7'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

Another advantage of the Smart 7 spectrograph is its ability to maintain a lot more history than ever before. In older versions of Smart, the number of "slices" of spectrograph history that you could maintain was limited by how many pixels on the screen that could be allocated to the graph. In Smart 7, the limit is more a matter of how much memory you want to allocate.

The amount of spectrograph history that you can display is still limited by screen resolution and 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 it 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.

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.

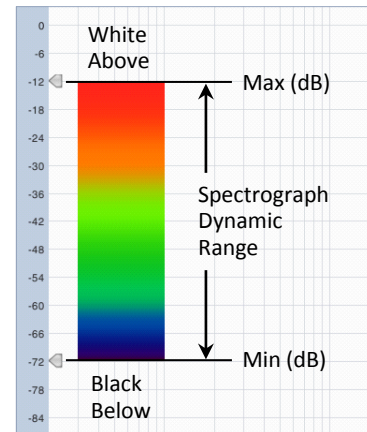


Figure 48: Spectrograph dynamic range and color mapping.

Spectrum Options

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

General Settings

There really is not much to talk about here. The only setting in this section is the *Data Window*, used to precondition time-domain signals before performing an FFT to find their spectrum. The default setting is *Hann* and unless you have some good reason to want to change it, that is a good place to leave it. A discussion of why one might want to use a different window function for real-time measurements, or not use a window function at all, is a little beyond the scope of the document. Note that this setting applies only to un-banded (narrowband) spectrum measurements. For fractional octave banded measurements, Smart always uses a Hann window.

Graph Settings

Frequency Scale – The frequency scale selector sets the frequency scale and grid ruling options for the RTA display. Actually, there are only two scaling options: linear (*Lin*) and logarithmic. All of the others 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 1/3-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.

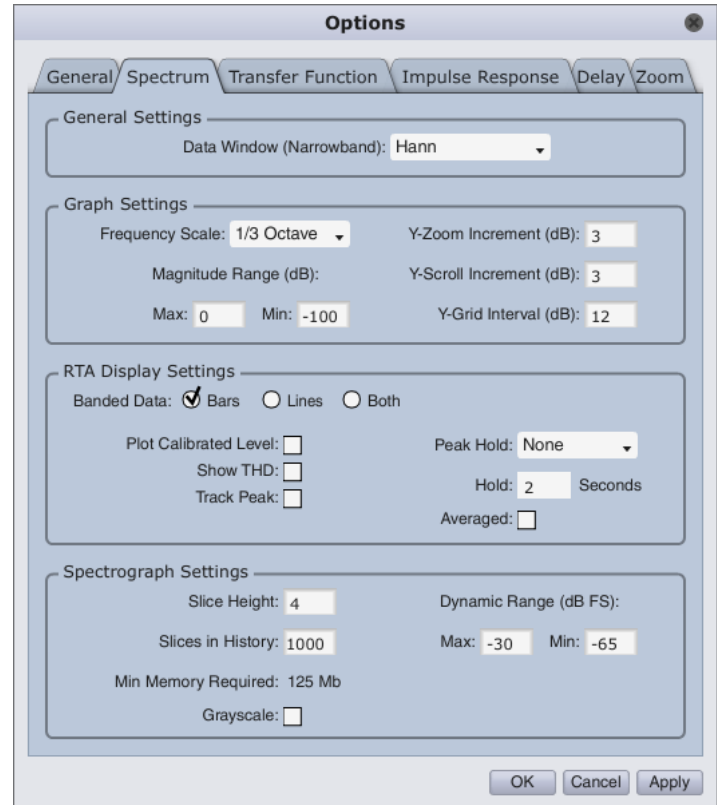


Figure 49: The Spectrum options dialog tab.

Chapter 5: Single Channel Measurements

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 [↑] or Down 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.

Peak Hold displays a line or bar segment at the peak magnitude measured for each frequency point on the RTA graph. There are three possible settings for peak hold:

- *None* turns peak hold off.
- *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.

Hold – this field sets the 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 calculates the memory requirements for the specified spectrograph history size based on the current FFT size selected for spectrum measurements in *Measurement Config*.

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

Dynamic Range (dB FS) sets the upper and lower (Min/Max) 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-through 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. If you already know your way around Smaart then just skip over the parts that you already know.

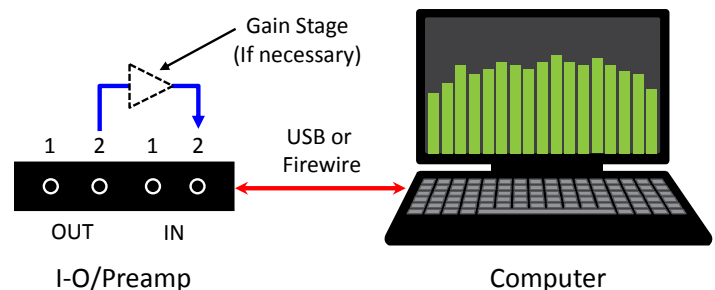
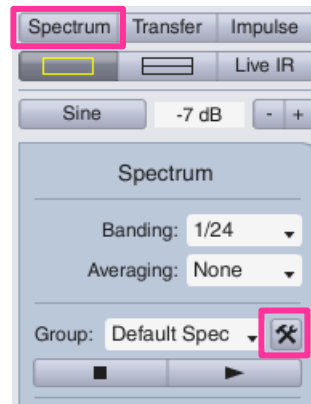


Figure 50: Loopback setup for distortion and overload example.

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 in the display control section of the control strip to set the graph area to a single graph pane and the graph type to RTA. If you don't already have a spectrum measurement set up for the input channel that you are using on your audio I-O device, click the little hammer and wrench icon on the control strip or select *Measurement Config* from the Options menu to open the *Measurement Config* dialog,



On the *Group* tab in *Measurement Config*, click the *New Spectrum Measurement* button. In the pop-up window, type a *Name* for the measurement, select the *Device* and input channel (*Ch*) that you are using, and then click *OK* to create the measurement. In the *Global Spectrum Settings* section of *Measurement Config*, make sure the *FFT* size is set to at least 8K or larger. Set *Banding* to 1/48 and *Averaging* to 8, then click *OK* to exit *Measurement Config*.

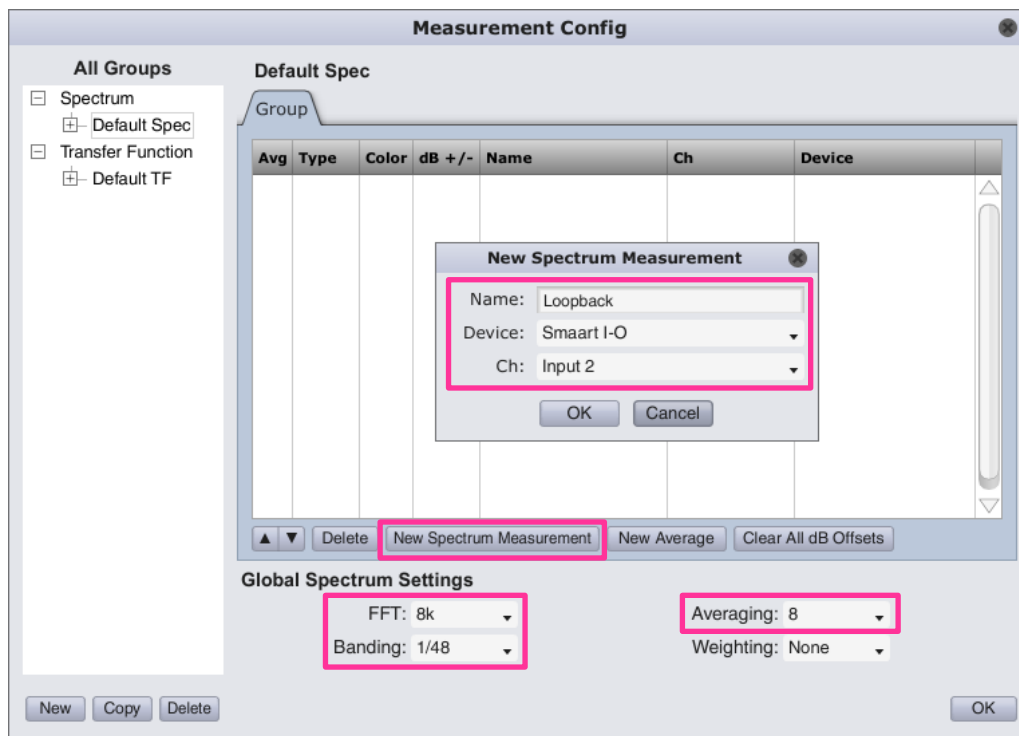


Figure 51: Measurement configuration for distortion and overload example application.

Back in the main *Smaart* window, you should now see a control block for your new measurement in the lower portion of the control strip. Click the run (▶) button for the measurement to make sure it works. All we care about at this point is that the run button turns green and *Smaart* doesn't throw any error messages. If there's any problem, double-click in the background of the measurement control block to open its tab in *Measurement Config*, and make sure you selected an input device that is currently connected to your computer. You can change it right there if necessary.

Now, click in the center portion of the signal generator controls (the output level readout) on the control strip or press [Alt/Option]+[N] on your keyboard to open the Signal Generator control panel.

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 window. The signal generator on/off button on the control strip should now say *Sine* (instead of Pink Noise). Go ahead and click it with your mouse or press [G] on your keyboard to start the signal generator.

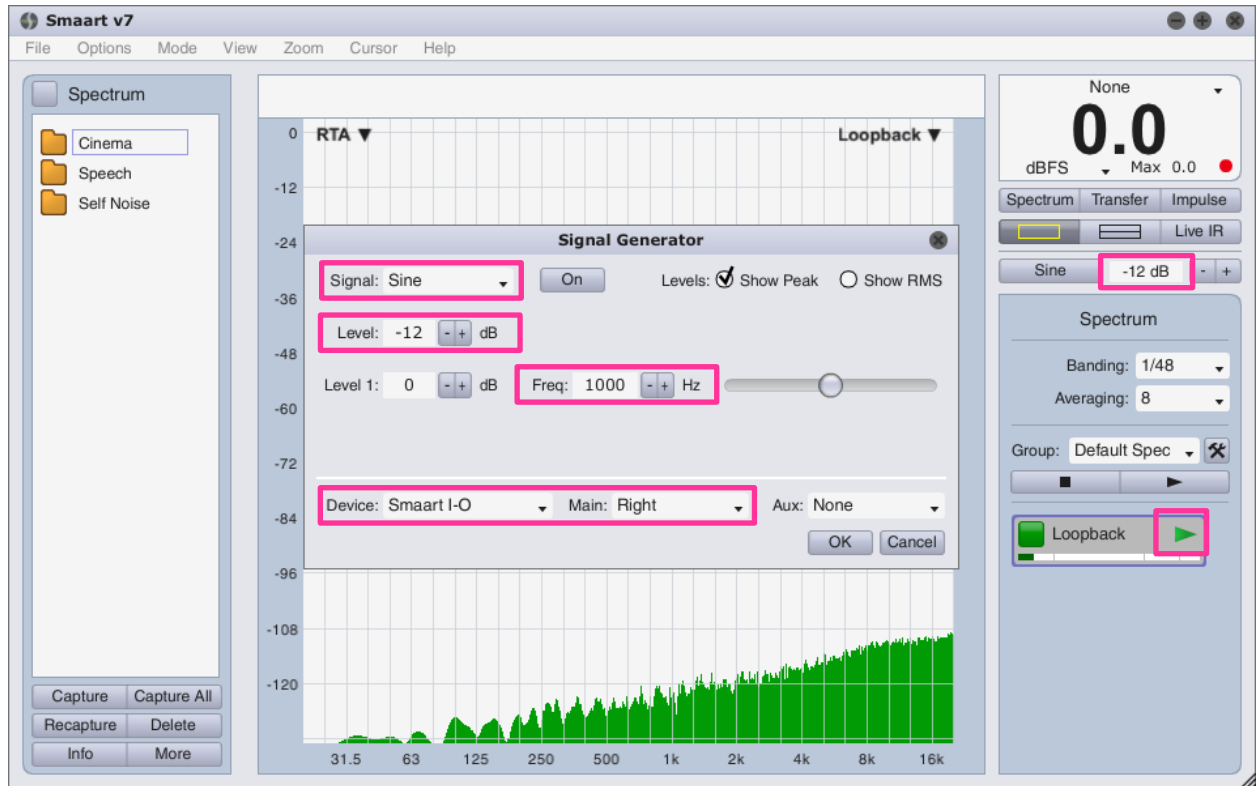


Figure 52: Signal Generator setup for distortion and overload example application.

If the measurement that we set up is not still running, click its run button to turn it on and then adjust the signal level until it's running 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 now see a nice clean spike on the RTA display at 1000 Hz like the one in Figure 53a. 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 we want to 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 53b. 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.

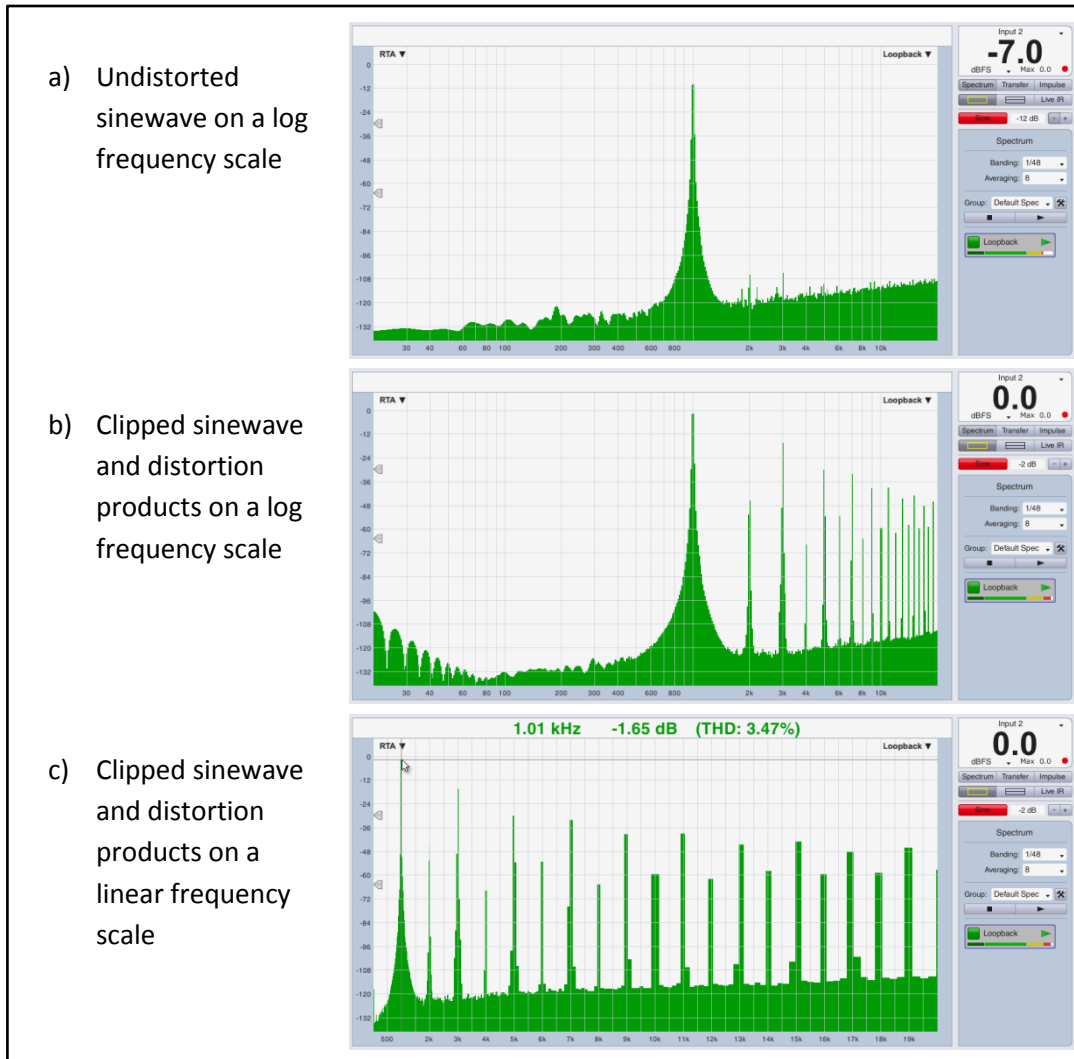


Figure 53: Measurement results for distortion and overload application example.

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 strip 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 a 1 kHz, Smart 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 things that we do with Smart.

Feedback Frequency Identification

For this next example, we need an actual sound system. It doesn't 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.

In Smaart, set up a spectrum measurement for the input on your I-O device that you're connecting to the mixer. See the previous example application (Distortion and Overload) for instructions on how to do this if you need them. Click the Spectrum button in the control strip, and then click the button with the split rectangle graphic in the next row down. Your graph area should then look like Figure 55, with an RTA graph on top and a Spectrograph below. Set the Banding selector in the control strip to the right of the graph to 1/24-octave, so we can see a little more detail on the RTA and Spectrograph.

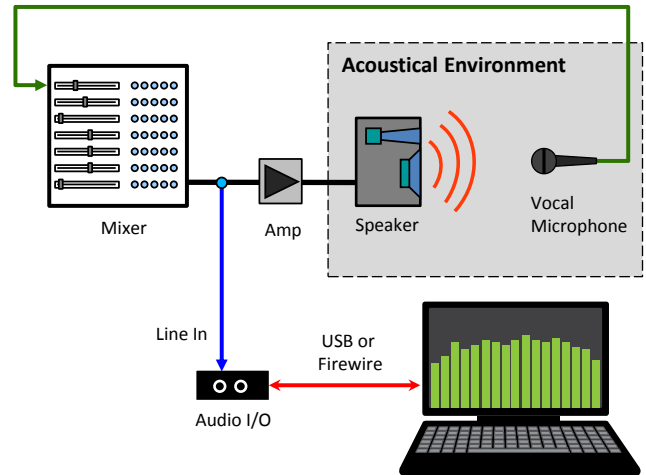


Figure 54: Measurement system setup for feedback study.

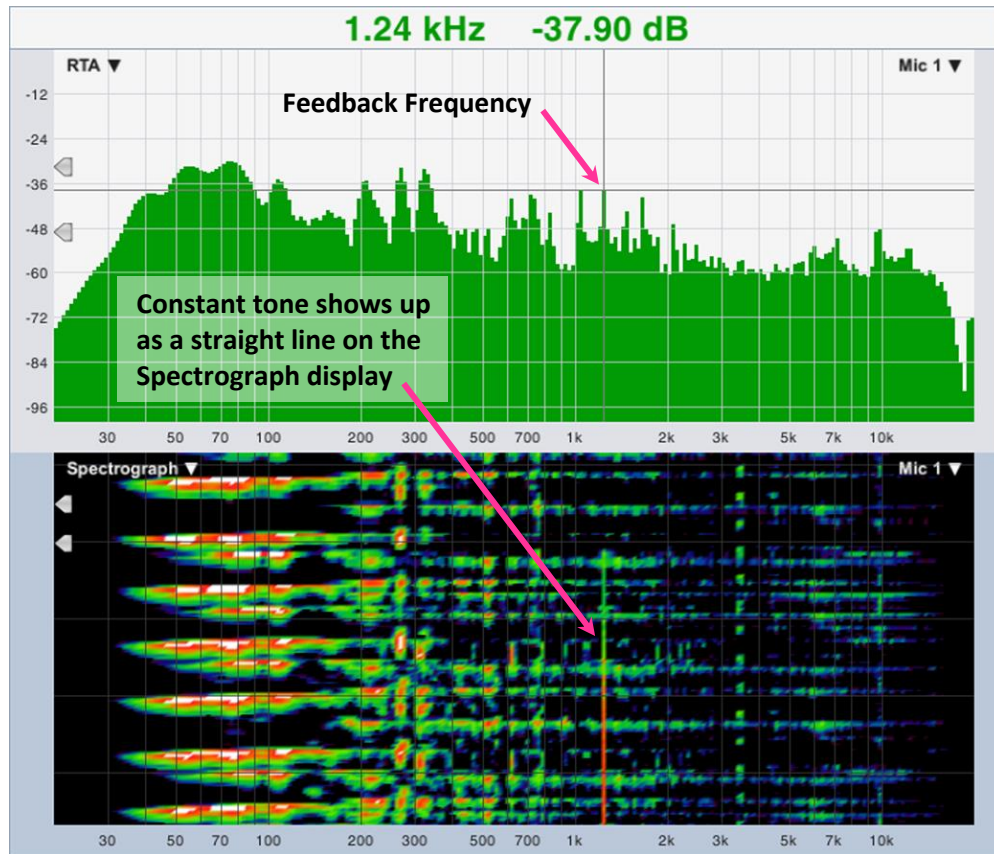


Figure 55: Feedback Study

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 **Error! Reference source not found.** 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 viewing the spectral content of their mixes, or in general viewing 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 build up between 160Hz and 220Hz, the spectrum analyzer set up you can help you see exactly what frequency is the main offender, and how much attenuation is needed to get back in line.

Examining Interaction Patterns with 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.

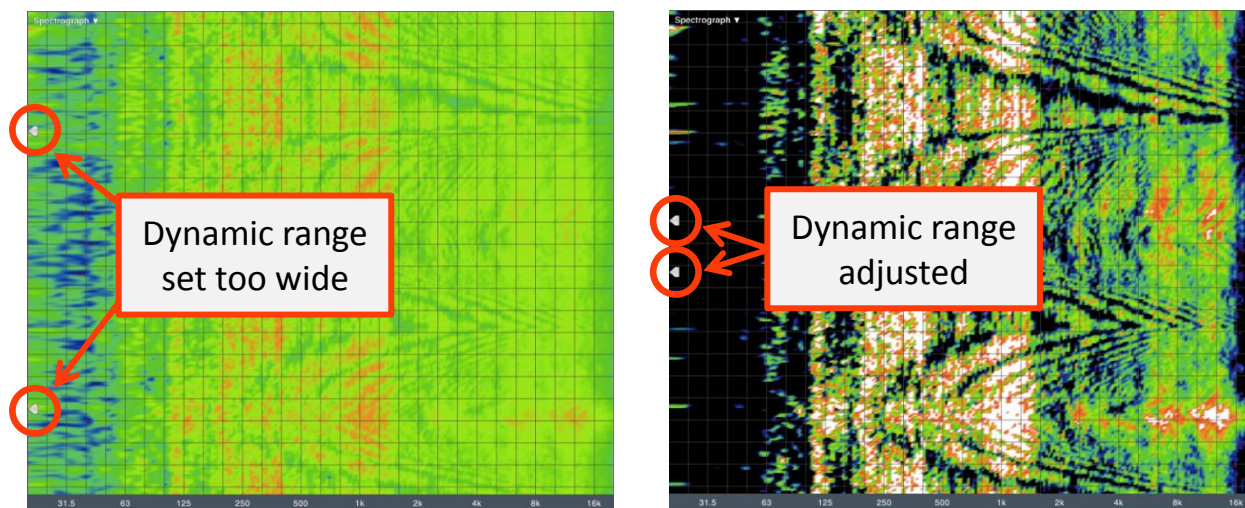


Figure 56: Spectrograph plot of comb filter interaction patterns.

Sound Level Measurement: SPL and LEQ

Sound pressure level (SPL) and Equivalent Sound Level (LEQ) are two ways of measuring and characterizing the loudness of sounds. Actually they are very much the same thing, the main difference being how the measurements are averaged and for how long.

Exponentially averaging sound level meters (SLMs) that measure SPL have been around for several decades. These rely on first-order lowpass filters for time integration, which are easily implemented in analog hardware – actually, just a resistor and a capacitor will do the trick. Most meters nowadays are digital but the specifications for exponential time integration have not changed. The time constant for the Fast integration setting is 125 ms. The Slow setting has a time constant of one second.

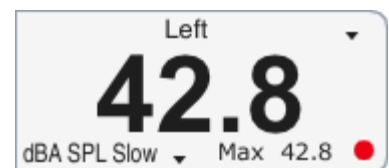
Integrating sound level meters used for LEQ measurement average linearly, over longer periods; often several minutes or even hours. LEQ is regarded as a better way of characterizing such things as background noise, environmental noise and dosages for sound exposure in very noisy environments such as manufacturing facilities, construction sites and rock concerts. Traditional SPL measurements use shorter integration times and are arguably still a better tool for measuring intermittent sounds or fluctuating sound fields.

Smaart can perform both SPL and LEQ measurements and given a good microphone and proper calibration, it can perform them very accurately. The DSP processes used for sound level measurements in Smaart were actually designed to the old Class 0 (laboratory instrument) specifications and exceed the requirements of current standards for Class 1 precision instruments. Unfortunately, there are no provisions in the current standards for certifying DSP processes independently of data acquisition hardware. The standards for sound level measurement equipment are written with hardware devices in mind and their test and certification procedures all pertain to the “complete instrument” – which in our case would include your computer, microphone(s), audio I-O interface and all cabling, in addition to the Smaart software. All this is to say that if you require standards traceability for your sound level measurements, you pretty much need a standards-compliant sound level meter.

Both SPL and LEQ are typically measured using standardized A or C weighting curves. The A weighting curve mimics the frequency sensitivity of the human ear at low levels and is also a good predictor of hearing damage from exposure to loud sounds. The A curve rolls off more drastically on high and low ends than the C curve, which simulates the sensitivity of human hearing at higher sound levels. There was also a B curve that split the difference between the A and C curves, but no one ever really used it for anything. The B curve has been dropped from the most recent standards but Smaart 7 still supports it.

Signal Level / Sound Level (Broadband) Meter

The large numeric readout in the upper right corner of the Smaart v.7 main window can be configured to function as a standard Sound Pressure Level (SPL) meter, an Equivalent Sound Level (LEQ) meter, or a peak signal level meter calibrated to digital full scale. The text labels above and below the large numeric value in the center of the display function as drop-down menus. The upper menu selects which audio



device and input channel drives the display. The one in the lower right sets the measurement type. Pressing the [K] key on your keyboard replaces this display with a clock, and pressing [K] again changes it back to a meter.

The selections for SPL and LEQ in the measurement type menu reflect the current averaging and weighting settings for these measurements selected in the Sound Level Options dialog (see below). For example, “dBC SPL Slow” means the meter is configured to display sound pressure level with C weighting and Slow exponential time integration. Similarly, “dBA LEQ 10” would mean equivalent sound level (LEQ) with A weighting and a 10-minute integration period.

The *Max* value keeps a record of the highest signal or sound level encountered since the measurement began or since the last reset. The little red circle (●) to the right of it functions as a reset button for the *Max* value in SPL and peak level modes. In LEQ mode, the reset button changes from red to green (●) once the buffer for the rolling LEQ average is fully populated, and clicking it will clear the LEQ buffer and restart the LEQ measurement.

Sound Level Options

Clicking anywhere in center portion of the meter opens the Sound Level Options dialog window. Here you can select weighting curves and integration times independently for both SPL and LEQ measurements, enable logging and specify a log file. The *Calibrate* button in the lower left corner opens the *Amplitude Calibration* dialog, where you can calibrate your input device(s) and microphone(s) for accurate SPL and LEQ measurement.

SPL and LEQ Settings

The *Type* selector in the *SPL Settings* section of the *Sound Level Options* dialog sets the integration period for exponentially averaged sound pressure level (SPL). The two options are *Fast* and *Slow*. Fast SPL averaging uses a first-order exponential time average with a 125 ms time constant. The *Slow* option has a one-second time constant.

Weight gives you a choice of A, B or C frequency weighting or no weighting (None), which could be considered equivalent to Z weighting, provided that your microphone and input device are flat within standardized tolerances from 10 Hz to 20 kHz. There are separate settings for SPL and LEQ weighting.

The *Minutes* field in the *LEQ Settings* section of *Sound Level Options* specifies the length of the LEQ averaging period in minutes. The value in this section is appended to the “LEQ” notation in the lower right corner of the signal/sound level meter when measuring LEQ. Typical integration periods for LEQ vary by application. The default in Smaart is 10 minutes, which is a fairly common choice for environmental noise measurements. Integration periods as short as 1 minute may suffice in situations where sound levels are not expected to vary by much over time. Often, community noise regulations or

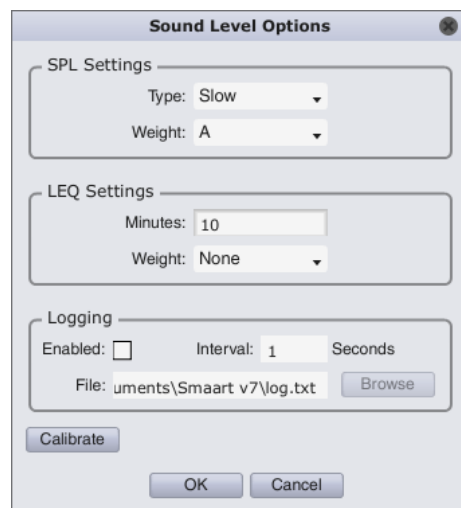


Figure 57: Sound Level Options dialog.

architectural specifications will specify the integration period to be used for LEQ measurements, so be sure to check any that may apply. In any case, the integration period for LEQ should always be noted in statements of measurement results.

Logging

The logging function for SPL, LEQ or signal level data simply writes the current meter reading to a text file at a specified interval. When enabled, it will keep writing to file until you tell it to stop or close the program. Logging is turned off by default each time you run Smart to prevent the log file from inadvertently eating up all the storage space on your computer, so in the event of a program crash or power loss you would need to go back in and turn it on again to restart logging if desired.

The log files are formatted in tab-delimited ASCII text, suitable for import into a spreadsheet, with one log entry per line. The format of the data depends on the measurement type. SPL and dB FS logs have columns for time and date of each entry, the meter reading in decibels and the measurement type. LEQ logs have additional columns.

Sound Level Calibration

To Calibrate one or more input channels for SPL measurement in Smart, press [Alt] + [L] on your keyboard or select SPL/LEQ from the options menu, then click the Calibrate button in the Sound Level options dialog. This opens the *Amplitude Calibration* dialog window shown in Figure 58. At the top of the calibration dialog are three drop-list selectors for *Input Device*, *Input Channel* and *Microphone*. Select the device and channel for the microphone that you want to calibrate, and if the input device happens to be a Smart I-O, then the *Microphone* selector becomes available as well.

Calibrating with a Sound Level Calibrator

In the realm of DSP, 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 sample word gets you max PCM amplitudes 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 dB scale. In order to relate that internal amplitude value to some number of volts or Pascals of pressure in the external world, we need to calibrate it to signal of known amplitude – and to do that with a microphone we need an acoustical signal of known amplitude. This is where a sound level calibrator comes in.

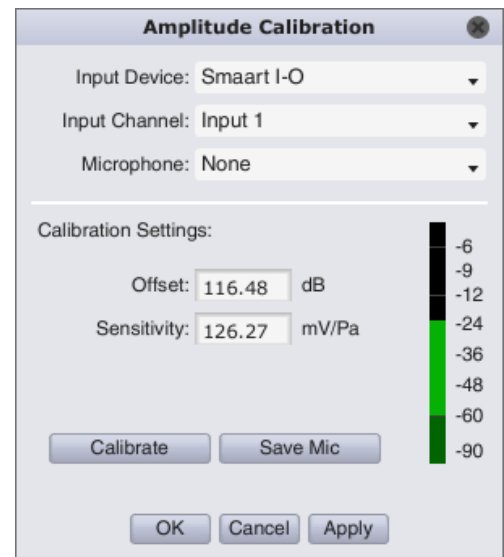


Figure 58: Amplitude Calibration dialog for SPL/LEQ calibration.

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

- a) Measuring the digital signal level of an input channel that is being driven by a signal of known amplitude, such as a microphone with a Sound Level Calibrator, and
- b) assigning the reference amplitude value to the measured Full Scale amplitude.

To measure the reference input signal level, you connect a microphone to the input channel that you want to calibrate, affix your sound level calibrator to the microphone, turn on the calibrator and adjust input gain to a desired level, then click the *Calibrate* button in the *Amplitude Calibration* dialog to run Smaart's calibration routine. Once calibrated you will need to leave the input gain set exactly where it is in order to maintain calibration (unless you are using a Smaart I-O), so a little forethought here can save you needing to repeat this procedure again later. Some things to think about are the loudest sounds you need to measure and the max SPL rating of your microphone.

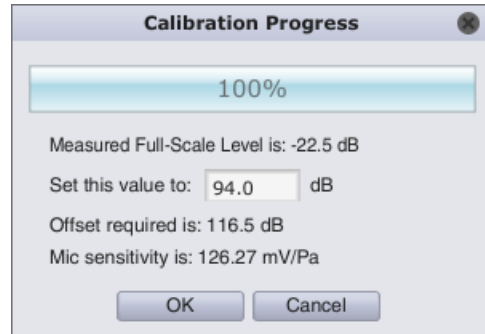


Figure 59: Calibration Progress dialog.

If your microphone is 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 might want to pick a lower number. 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 that's 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 more like -1 or -2 dB, just to make sure you don't clip the input during calibration.

On the lower right in the *Amplitude Calibration* dialog is a meter that shows you the peak full-scale signal level for the selected channel. 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. Smaart measures the input signal over a period of a few seconds and reports the full scale 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 the *OK* button.

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 is populated as well. If you are using a Smaart I-O, click the *Save Mic* button and give it a name – Smaart will select the new microphone automatically when you have done so. Otherwise we are done. You can click *OK* to exit the dialog or click *Apply* and then select another input to calibrate.

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 becomes possible to calculate the combined sensitivity of the preamp and microphone, provided that the microphone sensitivity is known.

Measurement microphones often come with individually measured sensitivity and frequency data, and so if your input device is a Smaart I-O and you know the sensitivity of your microphone in millivolts per Pascal, you can enter the number in the *Sensitivity* field. Smaart will calculate the required offset for SPL calibration when you press the [Enter] key to set the change. You can then save this as a named microphone by clicking the *Save Mic* button and giving it a name.

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 gives us the difference between the measurement and reference signals in both magnitude and phase. The measurement results represent the aggregate processing behavior of the system under test (SUT) as a function of frequency or time.

Dual-Channel Measurements: System Response Analysis

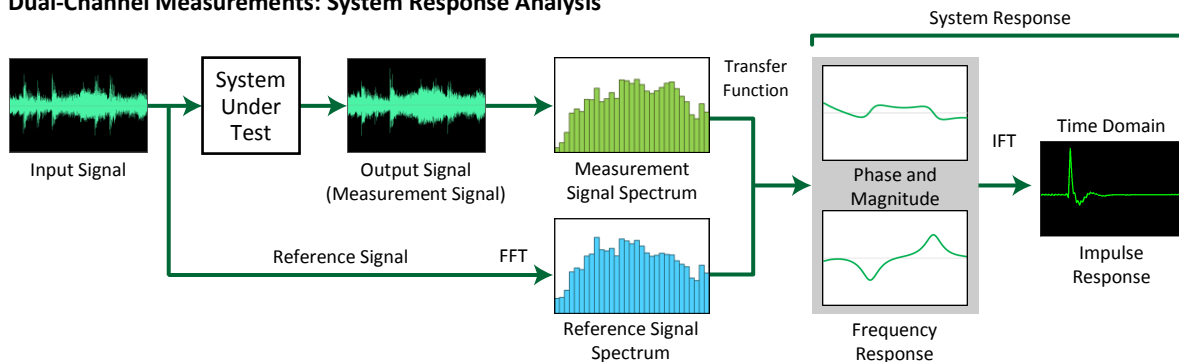


Figure 60: Block diagram of a transfer function and/or dual-channel impulse response measurement.

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 system under test, all the way through to the point where we picked up the resulting output of the system 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 we have discussed in some detail in *Configuring Live Measurements*. In this section, we will look more specifically at how some of those options 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 window (*Options* menu > *Transfer Function*), which we will be discussing in this section.

As was the case with Spectrum measurements, the line between the measurement and display functions gets a little blurry in places for practical reasons. Fractional octave smoothing, for example, is technically a display function that does not affect the underlying data, but we group it with the measurement parameter for usability 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 Strip

At a glance, the control strip for transfer function displays looks a lot like the one for Spectrum measurements. It says “*Transfer Function*” at the top instead of “*Spectrum*,” obviously. If you hover over the heading text 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 a *Banding* control there are separate smoothing controls for the magnitude and phase displays (*Phase Smooth* and *Mag Smooth*).

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

Another difference between the Spectrum and Transfer Function control strips are their live measurement controls. The *Group* 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.

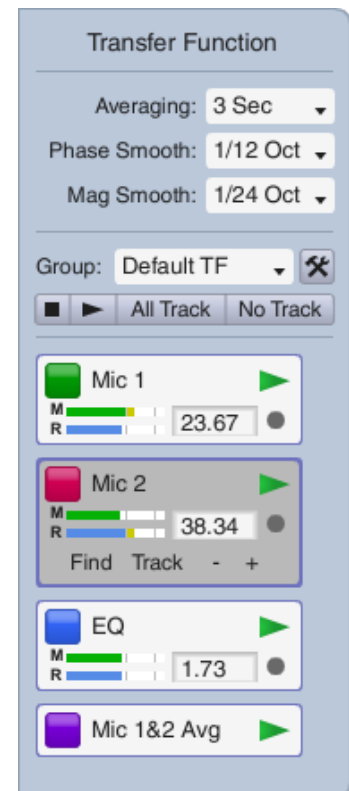


Figure 61: Transfer Function measurement control strip.

The individual live measurement control blocks for dual-channel transfer function measurements contain delay and delay tracking controls, in addition to the standard color-tile/hide button, measurement name and run/stop button, they include a delay field and two input level meters; one for the measurement signal (labeled “M”) and another for the reference signal (labeled “R”). Additionally, the control block for the active live measurement 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 routine, a Track button that toggles delay tracking on or off, and -/+ buttons that nudge the delay time setting up or down by one sample.

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 on the little hammer and wrench button next to the *Group* selector (see Figure 61) to *open Measurement Config*, we find that here too there are a few more global parameters than there are for spectrum measurement. When you look at individual transfer function measurement tabs you can also see that there are more localizable options than there are for spectrum measurements.

FFT Size

The FFT selection for transfer function measurement can be set globally or at the individual measurement level, rather than just globally for all measurements as was the case with spectrum measurements. The *FFT* selector for transfer function measurements includes the same selection of power-of-two FFTs offered for spectrum measurements, plus one additional option called *MTW*.

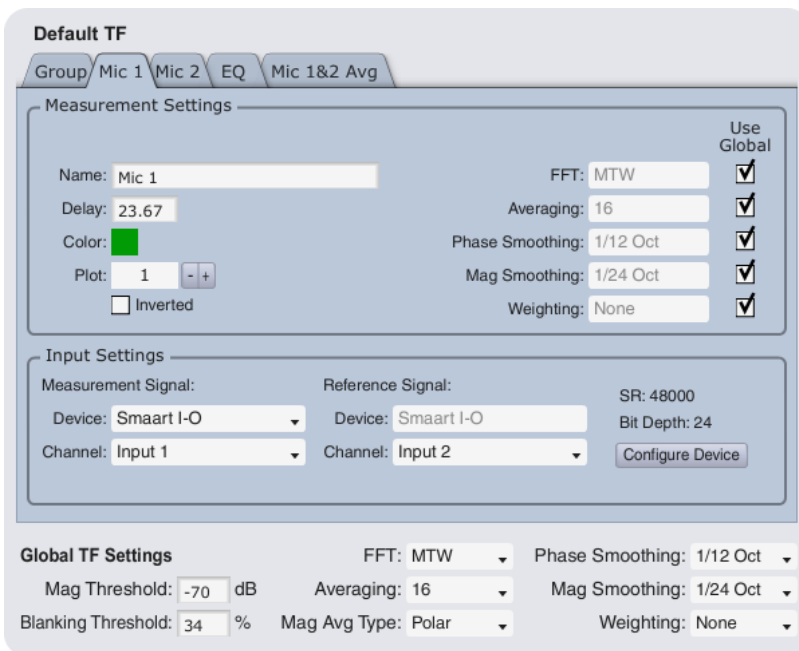


Figure 62: Transfer function measurement parameters.

MTW stands for multi-time-window. This is the default FFT selection for transfer function measurements and for the large majority of system tuning applications, there should 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 is that MTW 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 better computational efficiency – and with nearly 20 times fewer frequency data points than a 32K FFT it's 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 through digital devices, intentional alignment delay, and of course in the case of acoustical measurements, 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.

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 strip 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 strip – the measurement must be running in order to select it as the active measurement. Make sure that the signal source being used to excite

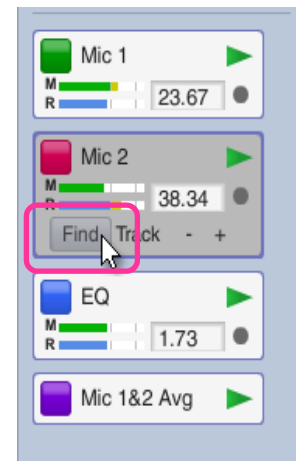


Figure 63: Delay Finder button for the active transfer function

the SUT is turned on and the input levels for the measurement are running at reasonable levels, then click the Find button that appears below the meters in the active measurement control block. The *Delay Finder* window appears, Smart 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.

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*, Smart 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 the 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 Smart 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 and it is possible that one might provide a better answer than the other.

Delay Tracking

Smart'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.

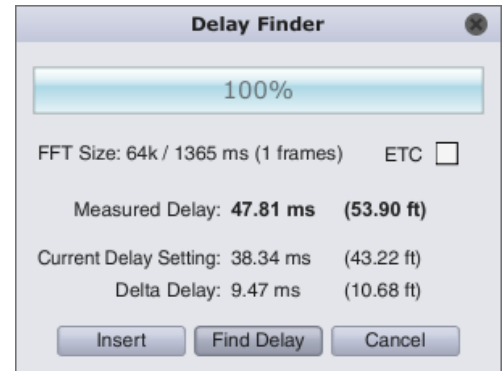


Figure 64: The automated delay finder.

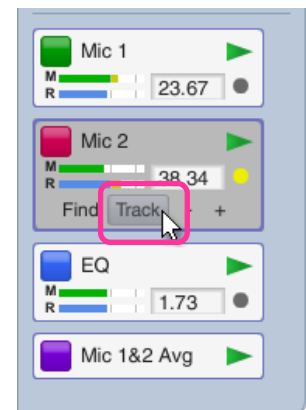


Figure 65: Delay Tracking button for the active transfer function

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.

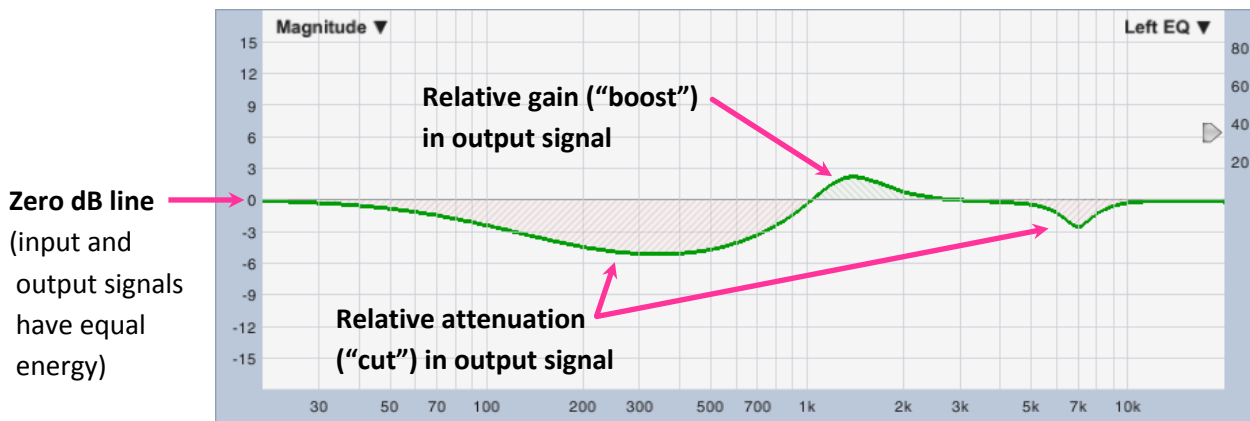


Figure 66: The transfer function Magnitude graph.

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 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 both 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 if the reference and measurement signals for the transfer 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.

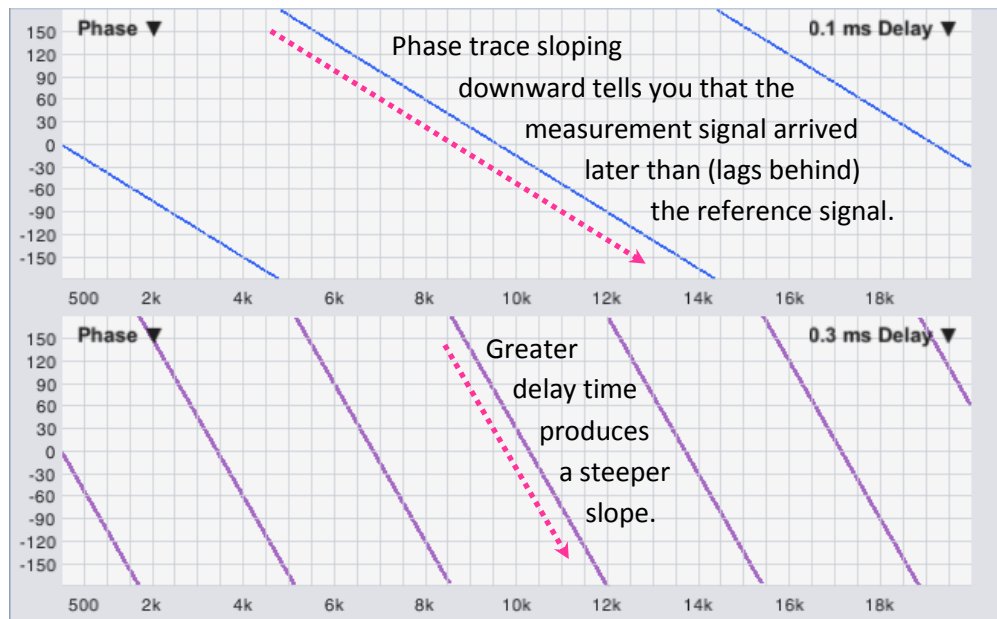


Figure 67: Uniform (linear) delay on a linear frequency scale.

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.

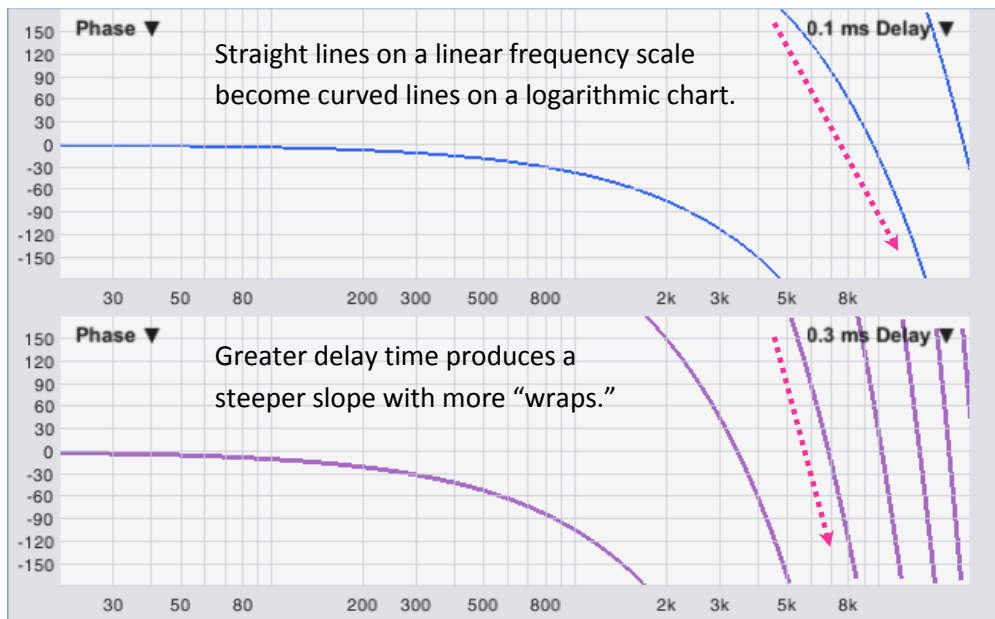


Figure 68: Uniform (linear) delay on a logarithmic frequency scale.

For example, a 10 Hz sinewave has a cycle time of 0.1 seconds, or 100 milliseconds ($1 \div f$, 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 $1 \div f$ 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 or click on a phase trace and drag it up or down to change the range of the phase graph, it is analogous to gluing the tube back together and then slicing it again at a different point. This, incidentally, is also why Smart 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 real data and properly should not be there.

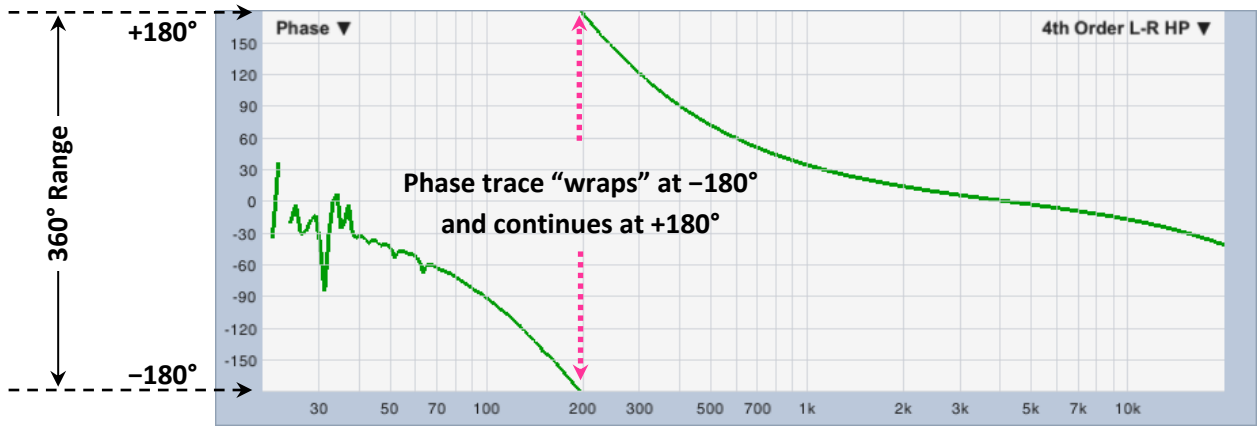


Figure 69: The standard "wrapped" Phase display.

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 Smart 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 – that 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.

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 produces exactly the same amount of

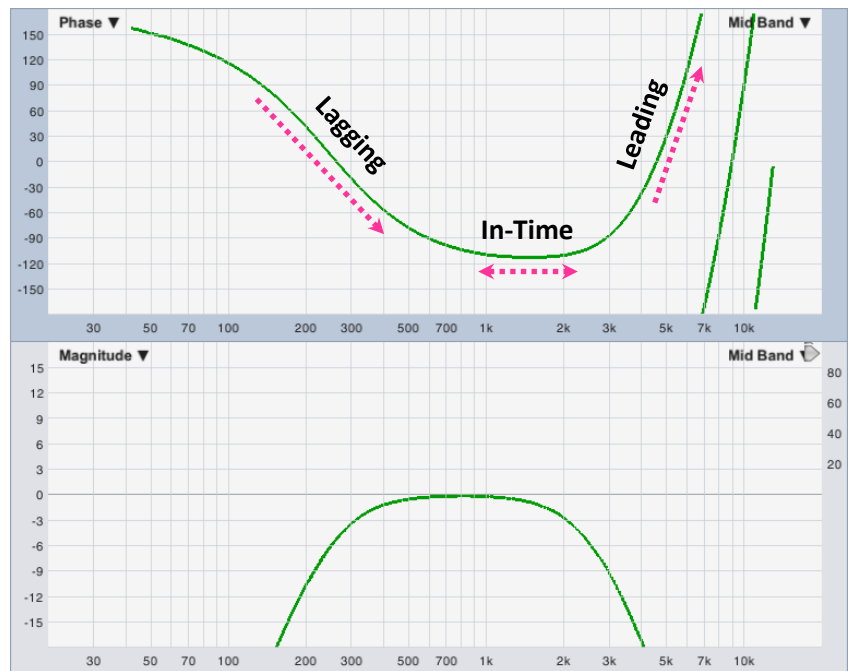


Figure 70: Phase and magnitude response of a 4th order Linkwitz-Riley bandpass filter.

phase shift forward and backward, so that the 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 right on 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 across the board. You have to pick a frequency range where you want the two functions to align. This is where the phase display comes in handy.

Comparing Phase Traces

When comparing phase traces, apply the up/down/sideways rule for leading, lagging or aligned in time:

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

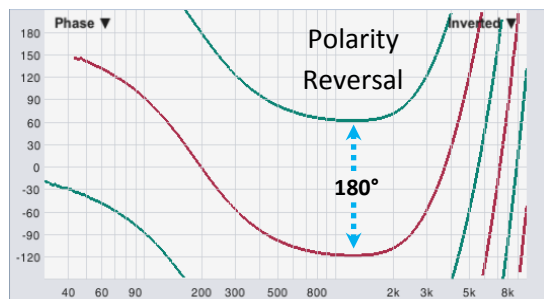
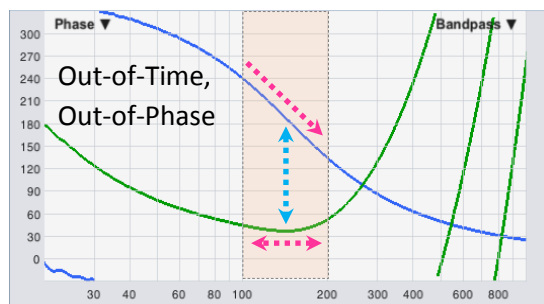
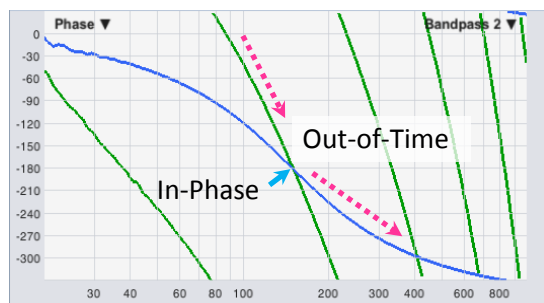
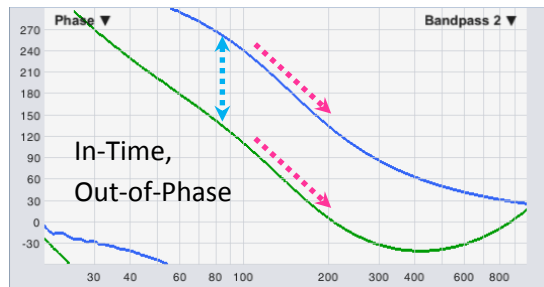
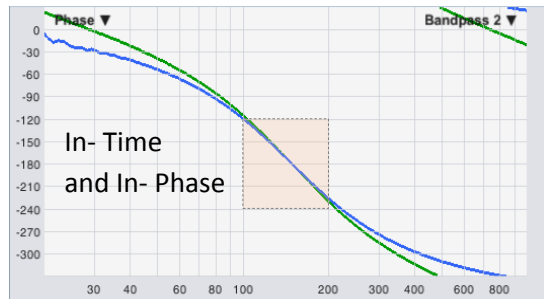


Figure 71: Comparing phase traces.

- Two traces with different slopes at 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's a *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 $\pm 180^\circ$. 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. It is like an analog clock with no hour hand, only a minute hand – it can tell you how much of the current hour has elapsed but it cannot tell you what hour of day or night it is.

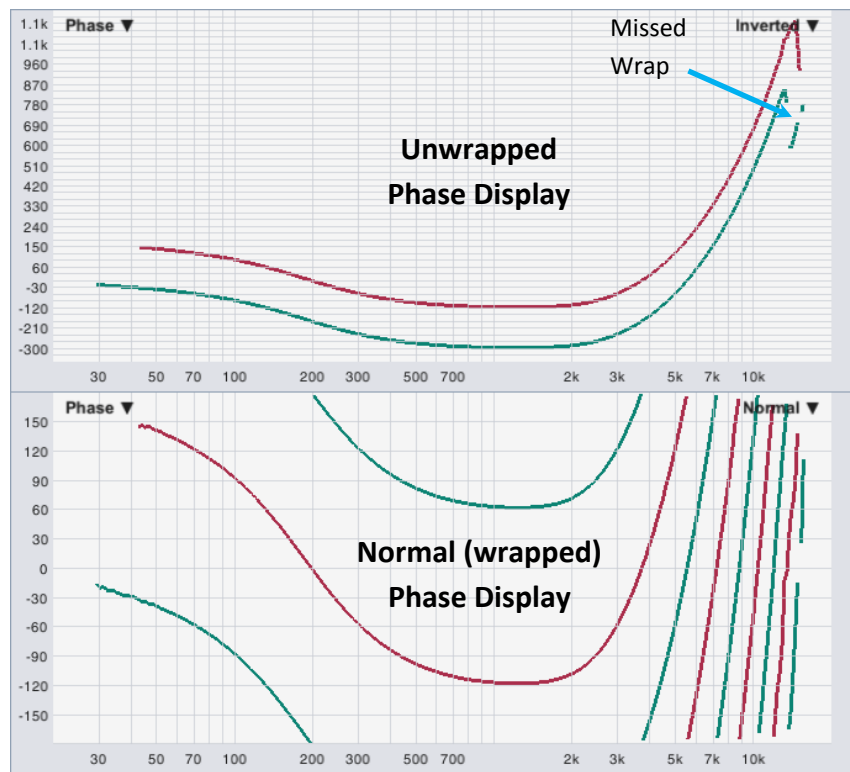


Figure 72: Phase on an unwrapped, versus wrapped (normal) phase display.

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 with no hour hand cannot really tell us what time it is over any timeframe longer than one hour, we *could* count the number of hours that have elapsed since we started watching it by keeping track 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 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 strip 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. 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.

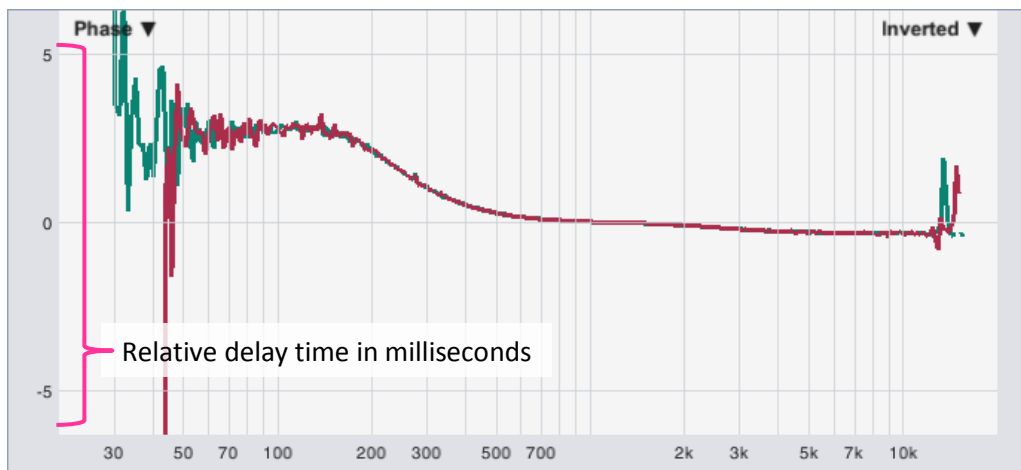


Figure 73: Phase shown as group delay.

To plot phase as group delay, click on the *Transfer Function* label above the active measurement controls on the control strip or select *Transfer Function* from the *Options* menu to bring *Transfer Function* options, 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.

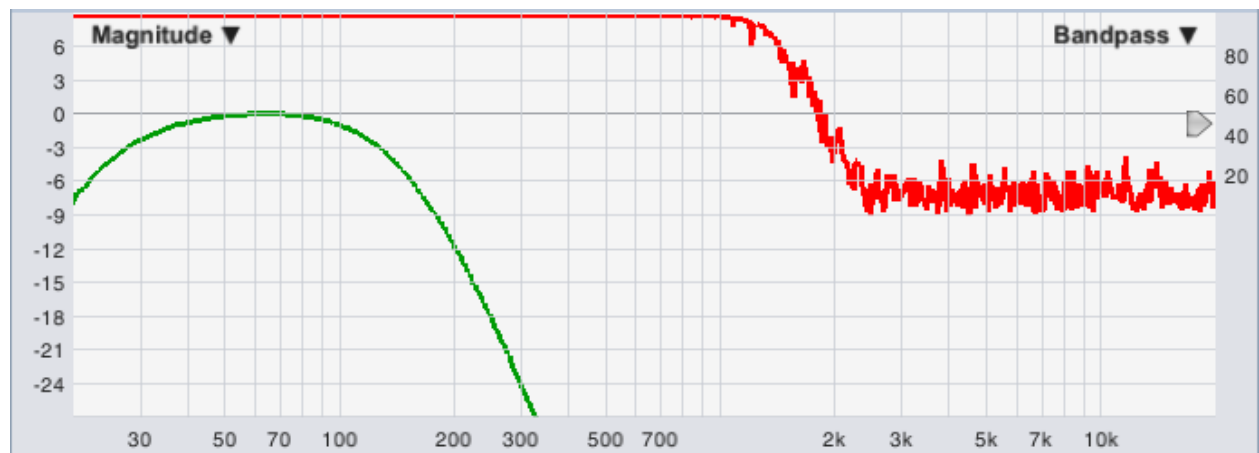


Figure 74: The coherence trace is plotted on scale of 0-100 in the upper half (or quarter) of the magnitude graph.

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.

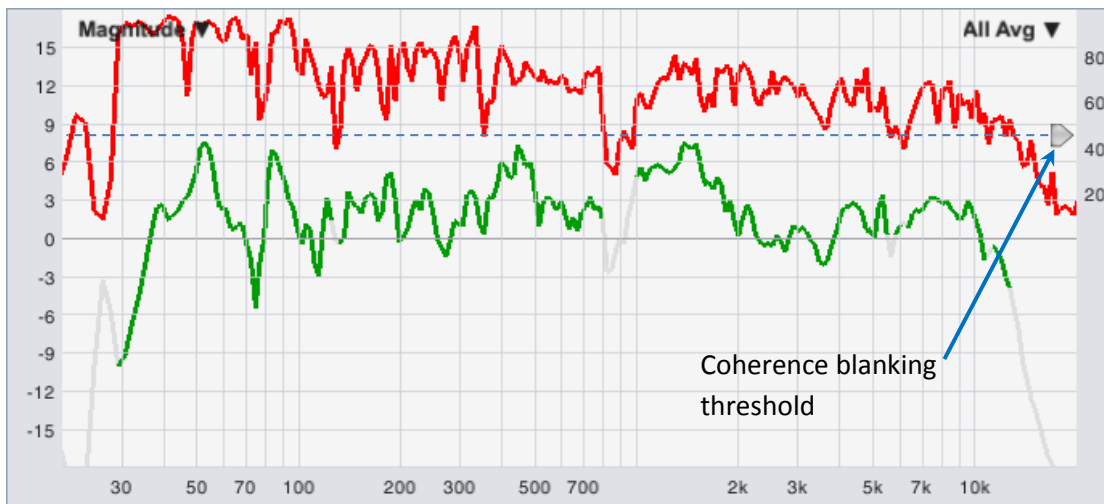


Figure 75: Coherence blanking.

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, *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.

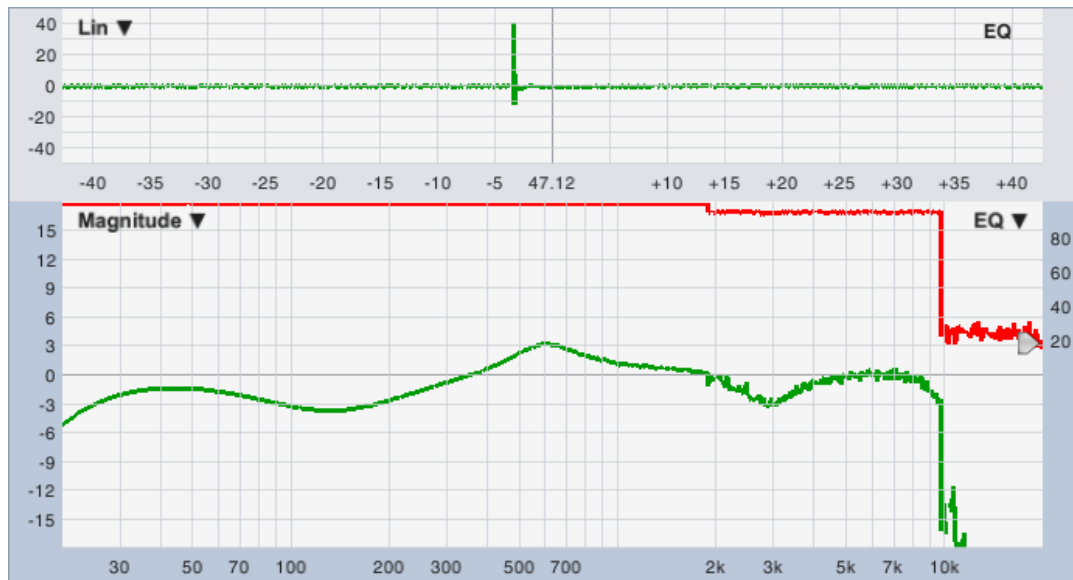


Figure 76: HF coherence loss in the MTW transfer function due to delay mismatch.

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.

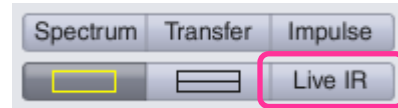
Reverberation

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 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 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 transfer function trace that does not include IR data.

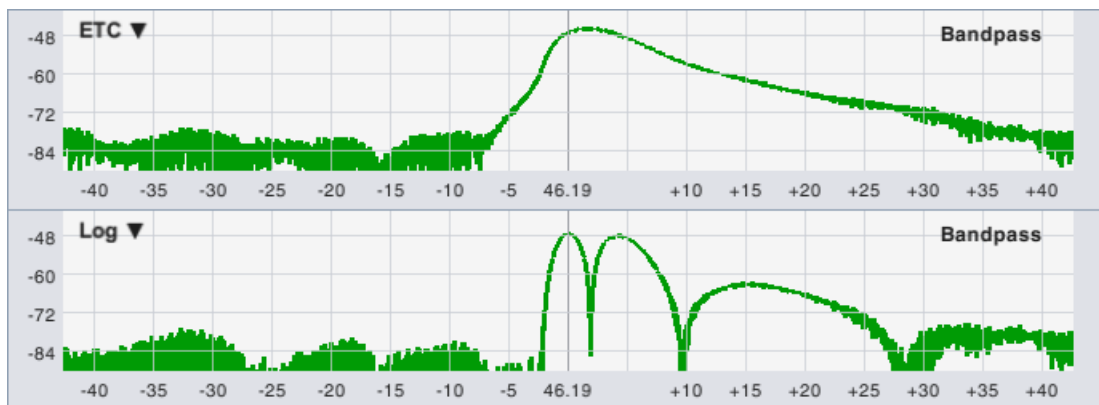


Figure 77: Log and ETC views of the impulse response of a low-frequency bandpass filter.

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) 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 strip 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 the 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 94, in the section covering coherence. Coherence blanking is one of several ways Smart 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 Smart 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, Smart leave 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 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, Smart 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 Smart 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, Smart 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 Smart 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, Smart 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, Smart 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 setting 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 strip in the main window, or you can select *Transfer Function* from the *Options* menu or press [Alt/Option]+[T] on your keyboard.



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 in this list 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 linearly-spaced vertical grid ruling.

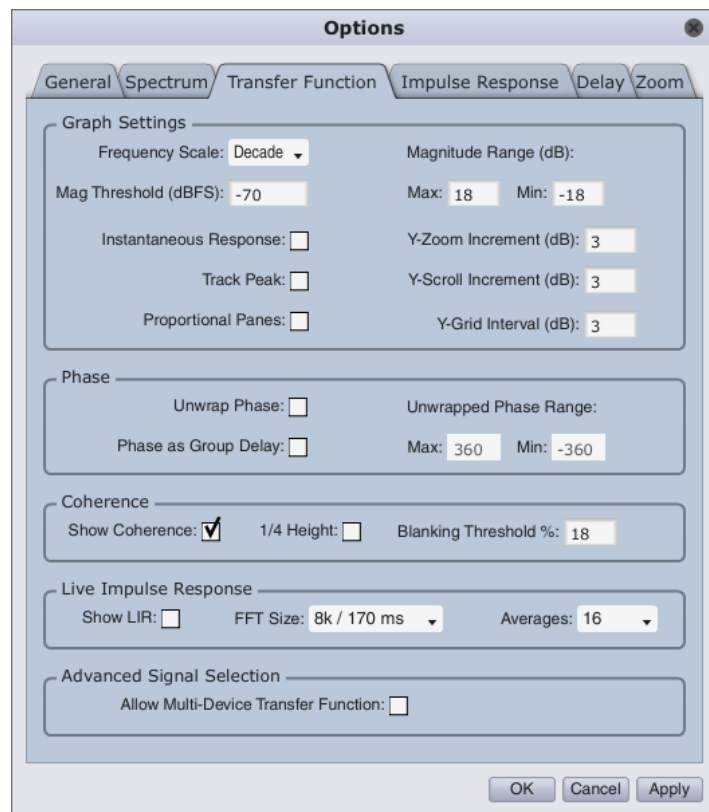


Figure 78: The Transfer Function options dialog page.

Mag Threshold (DB) 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.

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 in on the *y*-axis of the *Magnitude* graph. When 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 $\pm 180^\circ$ 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 $\pm 180^\circ$, 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.

1/4 Height Coherence: 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 strip in the main Smart 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 Signal Selection

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.

Application Example: Setting an Equalizer for a Loudspeaker

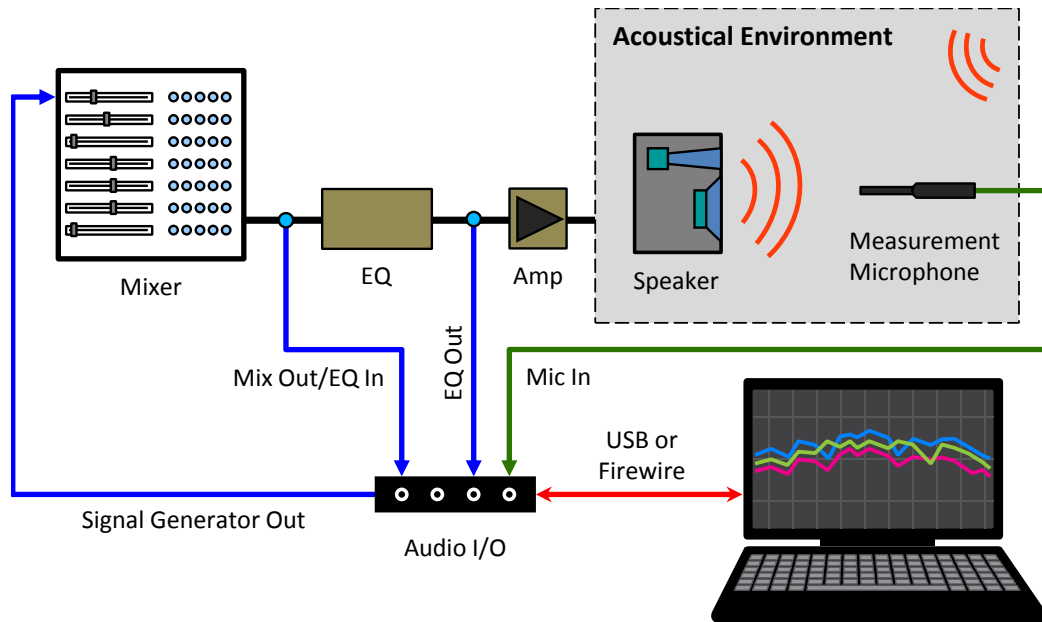
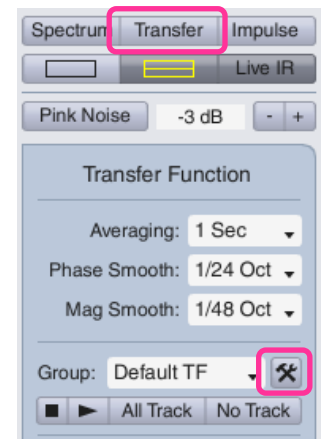


Figure 79: Measurement system setup for Setting an Equalizer for a Loudspeaker.

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 79. The setup shown enables us to simultaneously measure the equalizer (Mix Out compared to EQ Out in Figure 79) 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 the ball rolling open Smaart and click the *Transfer* button on the control strip 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 strip, since both graph panes contain transfer function graphs. Click the button with the hammer and wrench icon next to the *Group* selector in the control strip to open *Measurement Config*.

On the *Group* tab in *Measurement Config*, click the *New TF Measurement* button to create a measurement and name it “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. Click *OK* to create the measurement and exit the *New TF Measurement* dialog.



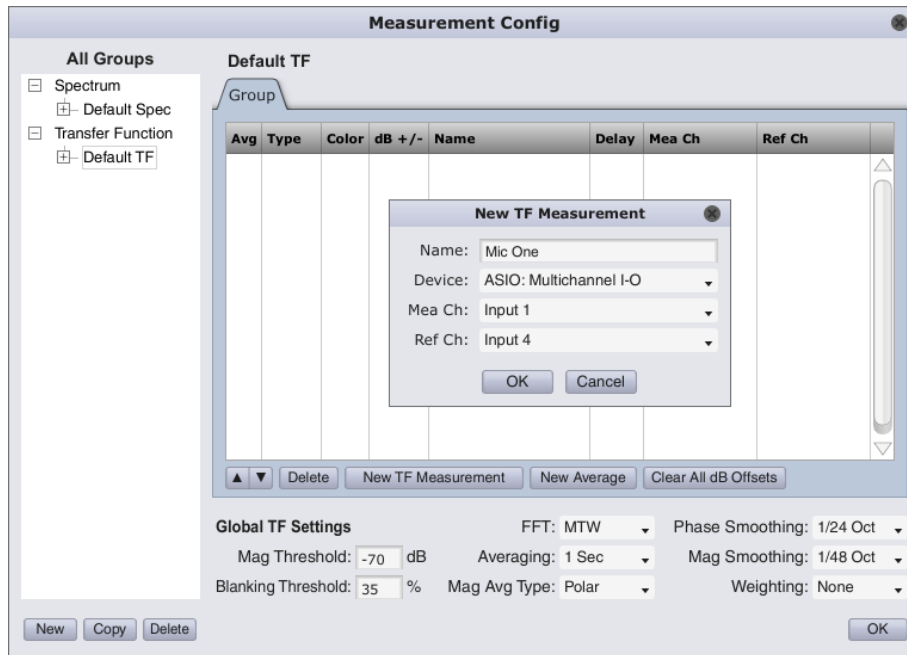


Figure 80: Creating a new transfer function measurement.

Repeat the steps above to create a second new measurement named “EQ.” Its reference signal channel should be the mixer output and its measurement channel should be the output of the equalizer.

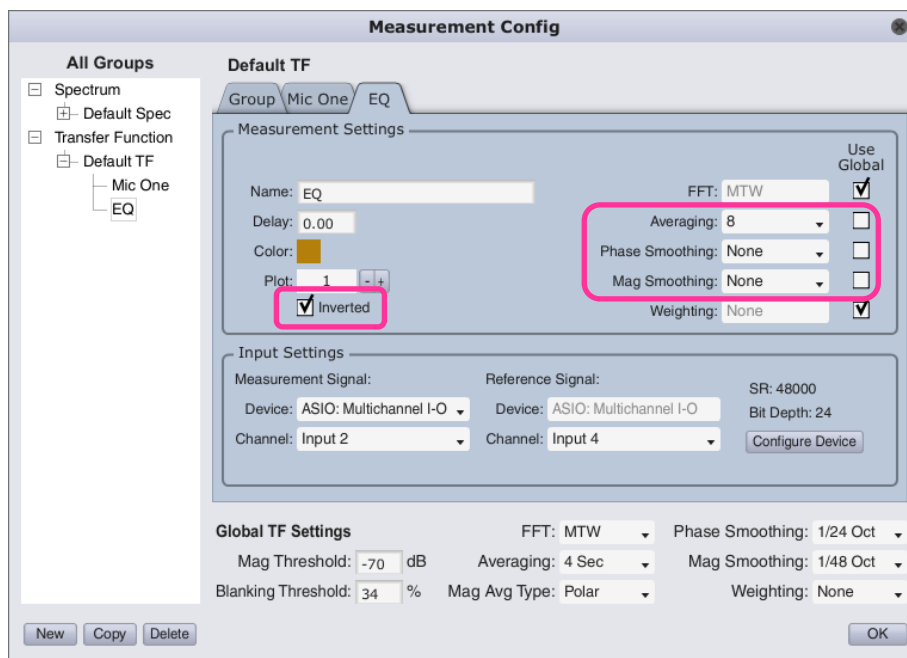


Figure 81: Measurement configuration for EQ measurement.

Notice that as you create new measurements, new tabs with the same names are added to the active transfer function group in *Measurement Config*. Click the settings tab for the EQ measurement to bring up its settings page and un-check the *Use Global* check boxes for *Averaging*, *Phase Smoothing* and *Mag*

Smoothing. Set *Averaging* to 8 or 16, and both smoothing controls to *None*. Click the *Inverted* check box (to display inverse EQ response), then click *OK* to exit *Measurement Config*.

Back in the main Smart window, start the signal generator with *Pink Noise* selected and adjust the output level to a comfortable volume. Use the gain controls on your 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 strip 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 82 – 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).

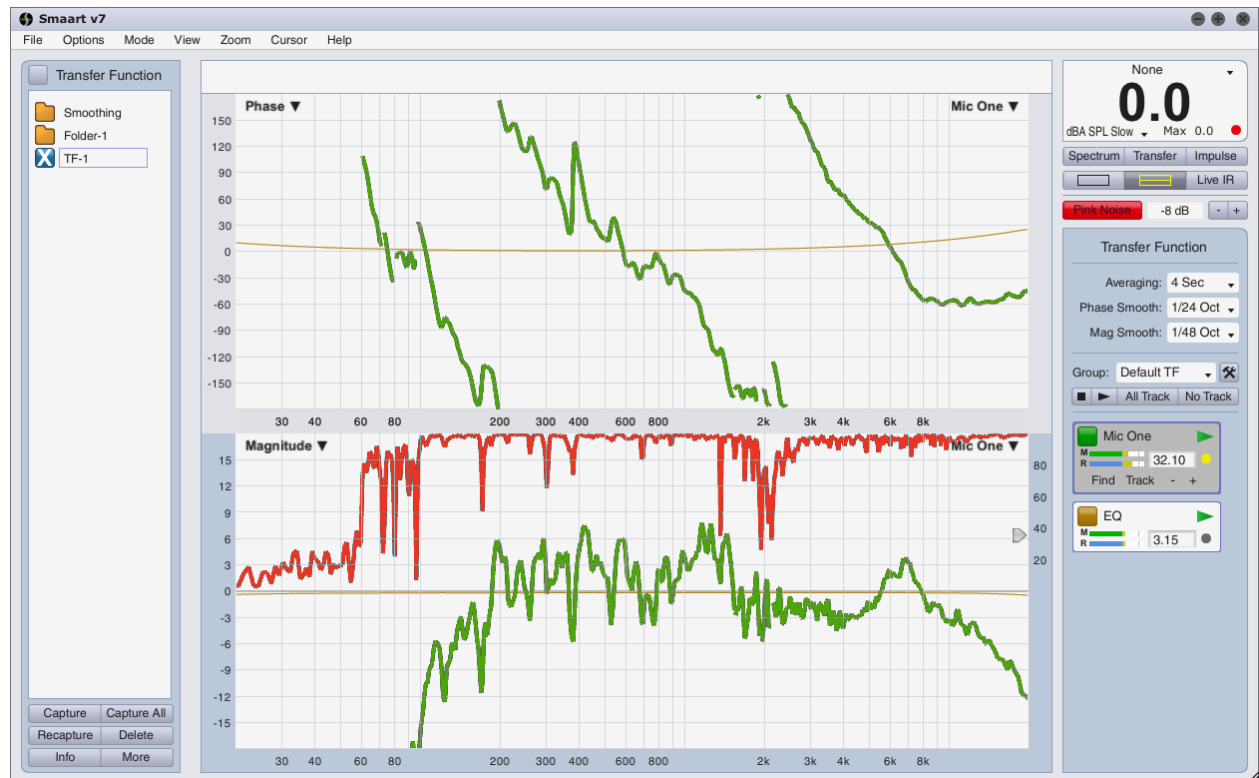
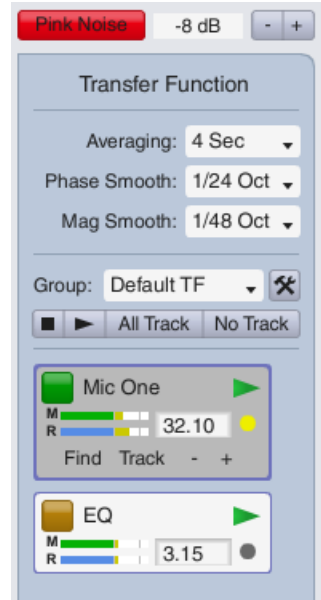


Figure 82: Loudspeaker measurement.

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 83*. Notice the effect that they have on the live measurement of the loudspeaker response (Mic One).

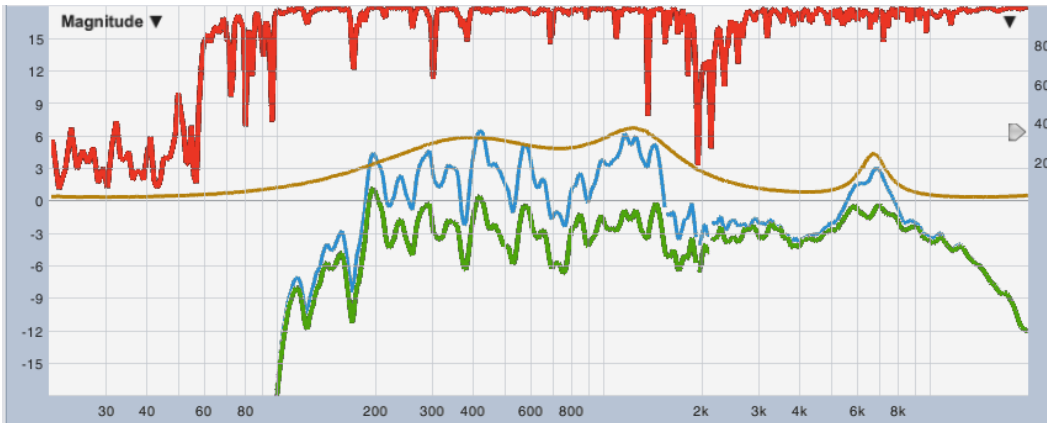


Figure 83: Transfer function measurement – initial loudspeaker response (Stored), EQ trace (inverted) and equalized loudspeaker response.

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. Smart 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.

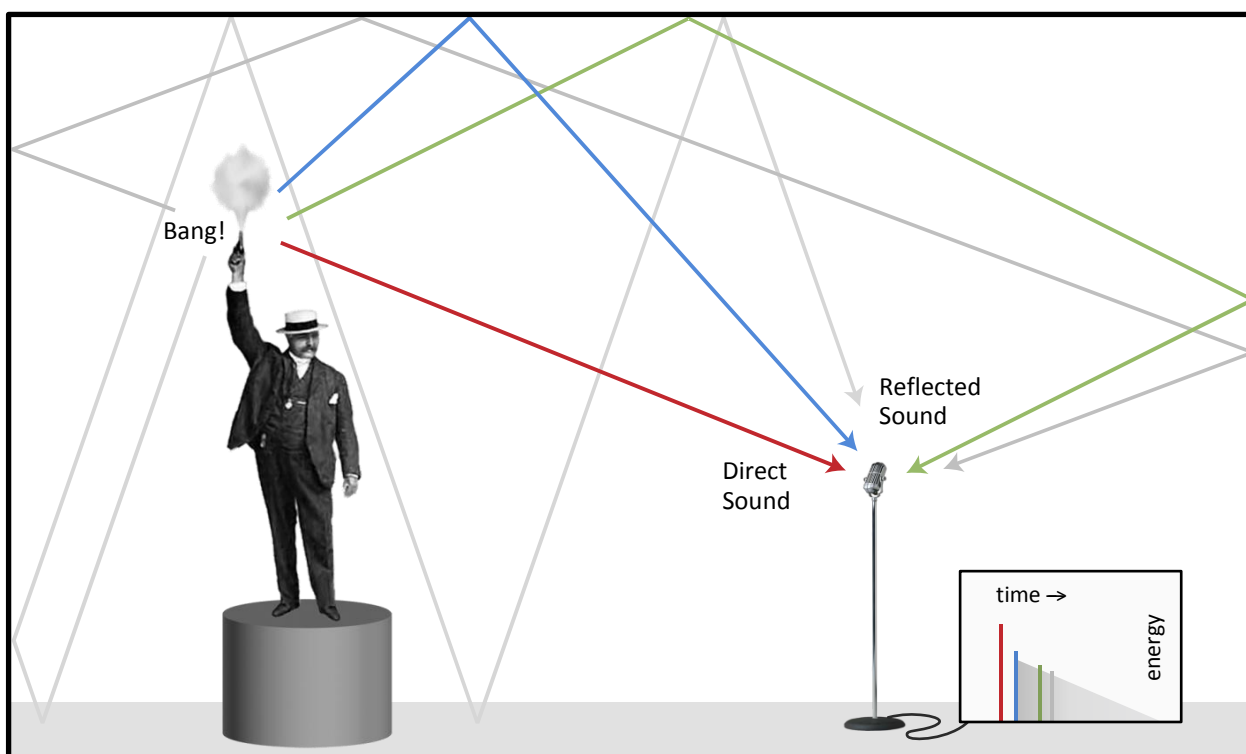


Figure 84: Conceptual illustration of an acoustical impulse. Sound from an excitation source arriving at a measurement position by multiple pathways, both direct and reflected. Here we see the path of direct sound from the source to the microphone in red, followed by a first order reflection in blue, 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 85 shows an acoustical impulse with its component parts labeled. Descriptions for each follow.

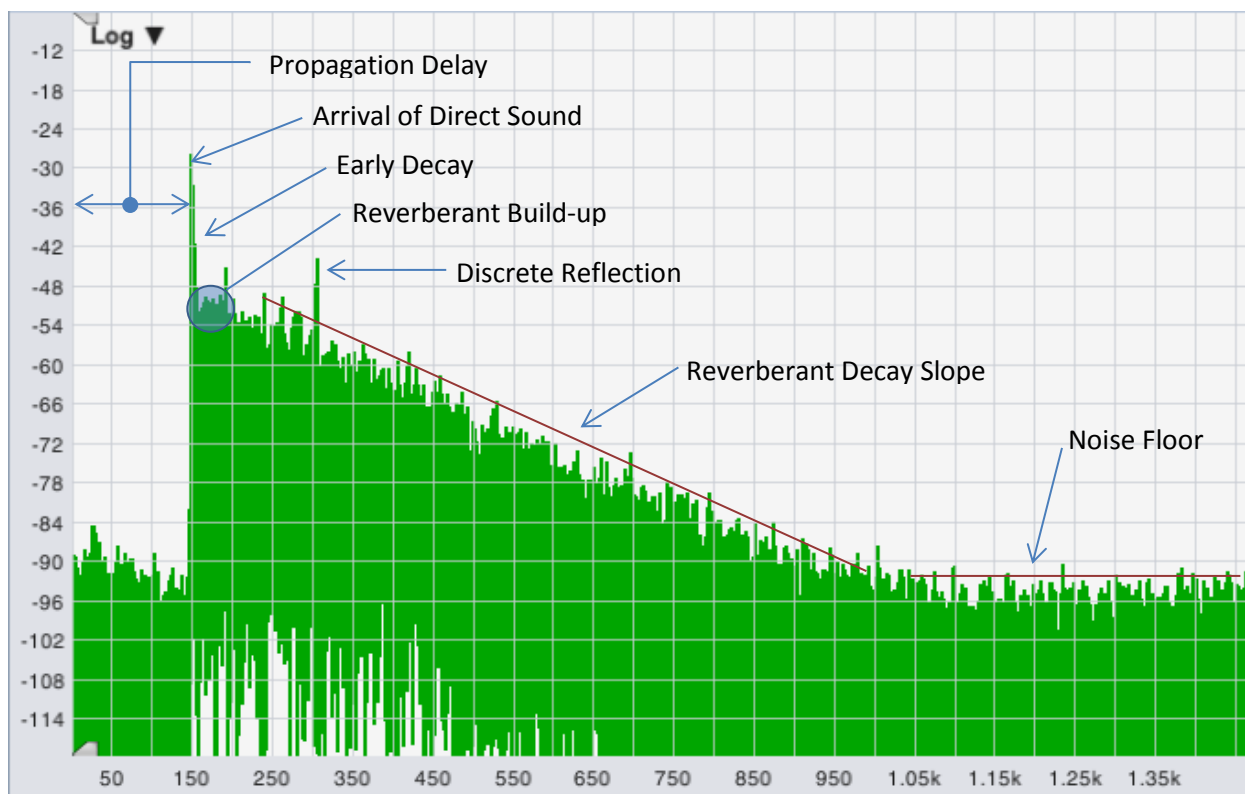


Figure 85: An acoustical impulse response with its common component parts labeled. This is a semi-log time domain chart with time in milliseconds on the x axis and magnitude in decibels on the y axis.

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 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, and electrical noise in the SUT and the measurement system, quantization noise from digitizing the signal(s) for analysis, and process noise 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 in a room sound 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

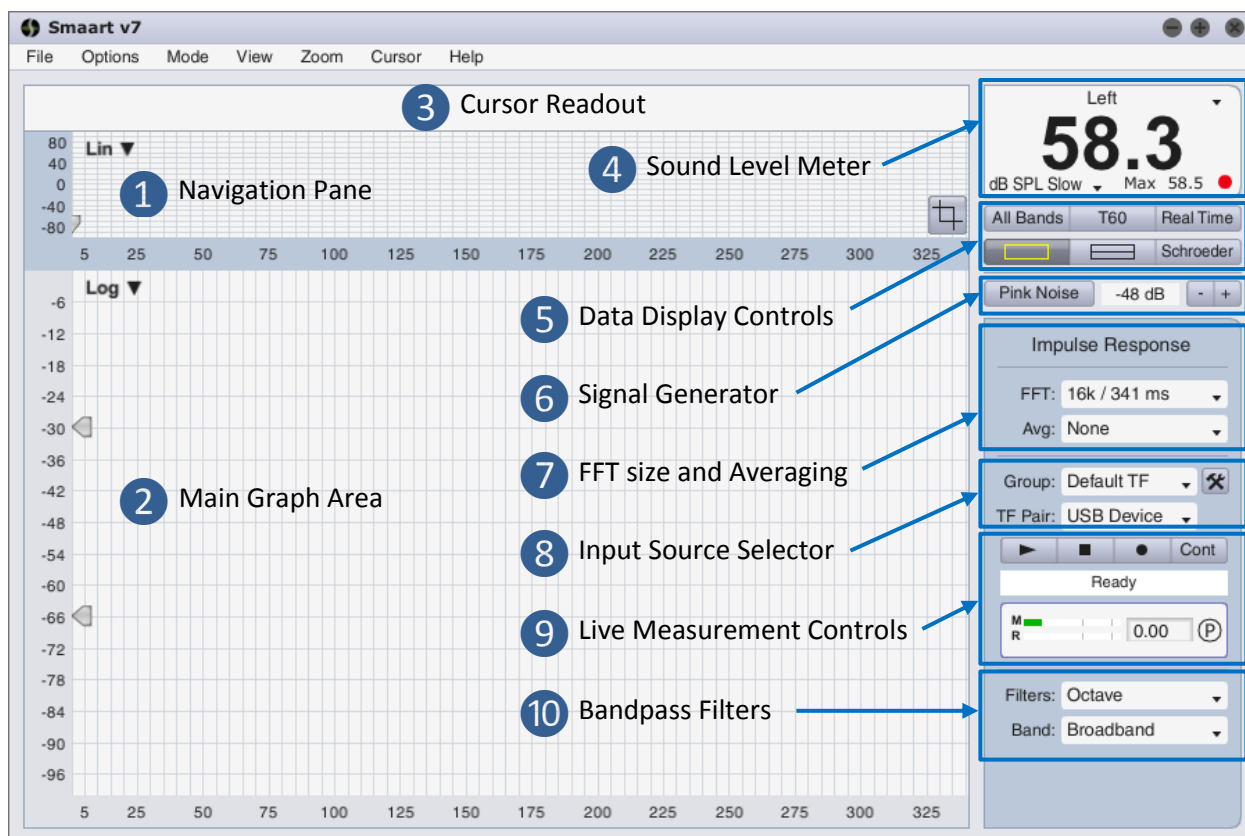
Early to late energy ratios are a direct measure of the sound energy arriving within some specified interval following the arrival of direct sound, vs the energy in the remaining part of the IR. These provide a more direct method of evaluating the relationship between beneficial direct sound and early 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 in 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 measurable 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.

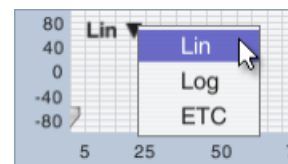
Chapter 8: Impulse Response Mode User Interface

If you already know your way around IR mode in Smart 7 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 Smart, select *IR Mode* from the *Mode* menu, press the “I” key on your keyboard or click the Impulse button that appears in the upper 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.



1 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 across this (Ctrl + click and drag on Mac) panel 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.

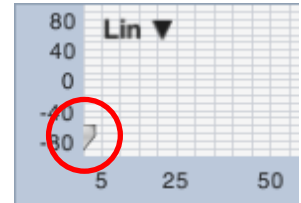


The selector control in the upper right corner of the navigation pane selects the graph type to be displayed in this area. The navigation pane is limited to time-domain graph types only (Lin, Log or ETC).

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.



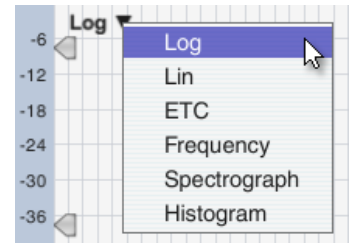
2 Main Graph Area

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's no multi-channel file support or optimization for working with files more than a few seconds in length – and of course a lot the IR analysis capabilities are irrelevant for other type of audio data.

3 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.

4 Sound Level Meter

The large numeric readout in the upper right corner of the Smaart v.7 real-time mode window can be configured to function as a standard Sound Pressure Level (SPL) meter, an Equivalent Sound Level (LEQ) meter, or a peak signal level meter calibrated to digital full scale. The labels above and below the large numeric value in the center of the display function as menus that enable you to select the input channel driving the display and the type of measurement you wish to perform. When operating as an SPL or peak meter, the Max value can be reset by clicking the red circle (●) in the lower right corner of the display. In LEQ mode it changes from red to green (●) once the buffer for the rolling LEQ average is fully populated and clicking it will clear the buffer and restart the LEQ. Using the [K] hotkey displays a clock with the system time in the SPL meter window. Clicking anywhere in center portion of the meter opens the Sound Level Options dialog window where you will find additional options for configuring and calibrating the meter.



5 Data Display Controls

The first group of controls in the control strip on the right side of the IR mode window is devoted to data display functions. Starting from the top left of the screen clip on the right:



- The All Bands button opens the All Bands table, where all of the quantitative acoustical metrics that Smaart can calculate are found.
- Clicking the T60 displays the level marker widgets used for calculating reverberation time and early decay time on Log IR or ETC plot(s).
- The Real Time button exits IR mode and takes you back to real-time frequency domain measurement mode. (In real-time mode this changes to the Impulse button that will bring you back to IR mode.)
- The two buttons labeled with rectangles divide the main plot area (#2) into one or two graph panes – 1 rectangle, 1 pane; 2 rectangles, 2 panes.
- The Schroeder button displays a reverse time integration curve on Log IR or ETC plots.

6 Signal Generator

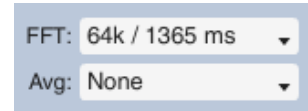
The Signal generator controls work the same way in IR mode as in real-time mode. The leftmost button displays the currently selected



signal type and turns the generator on and off. The +/- buttons on the right increase/decrease the output level. The center area displays the current output level and doubles as a button to open the signal generator control panel (also accessible by selecting Signal Generator from the Options menu) if you click it with your mouse. We will be talking more about signal generator options later on, as there are a few that are of particular interest in the context of IR measurements.

7 FFT size and Averaging Controls

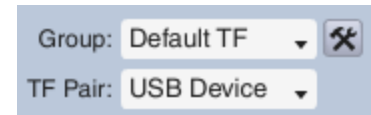
The FFT size and averaging controls together determine the measurement duration for dual-channel IR measurements. Notice that for each FFT size, the time constant is given along with the FFT size in samples. The FFT time constant, also called the time window, is the amount time it takes to record the 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 using random test signals. For deterministic IR measurements made using period-matched signals, the number of Averages is normally set “None”.

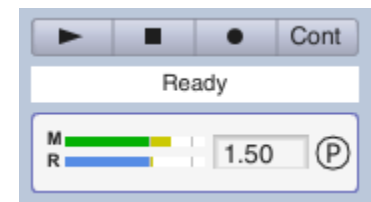
8 Input Source Selector

Dual-channel IR measurements in Smaart are essentially transfer function measurements with the addition of an inverse Fourier transform at the end. Any of the reference and measurement signal pairs that you have set up for real-time transfer function measurements may be used for IR measurements. For single channel IR measurements (see #9 below) only the measurement channel is recorded. If you have more than one Transfer Function measurement group, the Group control is used to select the group containing the measurement that you want to use.



9 Live Measurement Controls

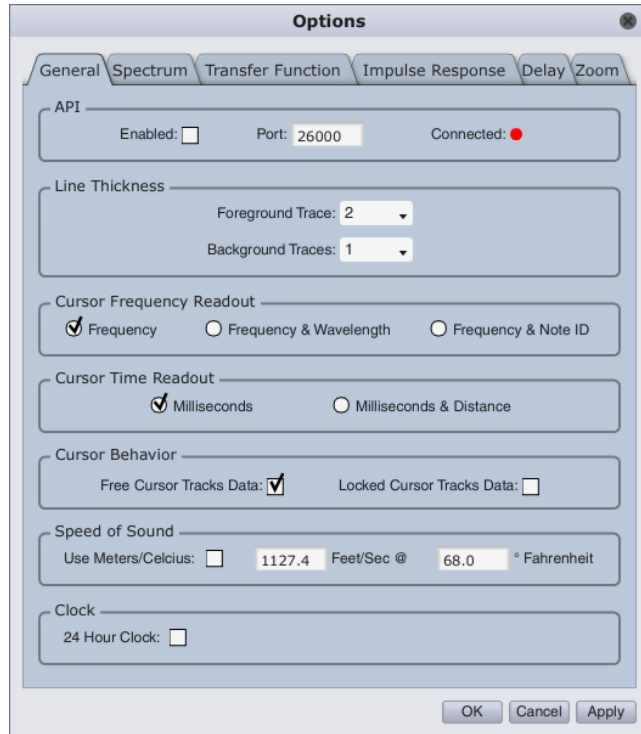
The live measurement controls in IR mode are analogous to a transfer function “measurement engine” in real-time mode, with a couple of extra twists. Starting from the top left, the buttons marked with a triangle (▶) and a square (■) start and stop a measurement. The button labeled with a circle (●) works like the record button on a tape deck or digital recorder but in this case it’s also 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 pair. The idea is that you would start recording and pop a balloon or fire your starter pistol (or whatever), then click the stop button to end recording and display your single-channel impulse response measurement.

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 measurement.

Underneath the recording controls there's a status bar to let you know what Smart is doing when a measurement is in progress, since IR measurements can take a while to complete. Below that are input level meters and reference signal delay controls for the selected signal pair. The "@" button sets the delay time for the measurement to the highest peak in the impulse response when clicked. If you are measuring using a random stimulus signal and the delay is not already set you would run the measurement once to get the delay time, then click the @ button, then run the measurement again. You generally don't have to worry much about the delay time with deterministic IR measurements. They work fine with the delay set to zero. To zero out the delay time, type a zero in the delay time field and hit the enter key.



10 Bandpass Filters

The broadband impulse response is useful for finding delay times but for most acoustical analysis purposes, the IR needs to be filtered into octave bands or occasionally, 1/3-octave bands. Smart 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 you have to do is select the filter set (that you want to use (Octave or 1/3-Octave) using the Filters selector in the lower right corner of the IR mode window and then choose the center frequency for the band that you want to analyze from the Band list.

Additional Options for IR Measurement

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 from the *Options* menu.

General Options Pertaining to IR Measurement

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 Smart 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.

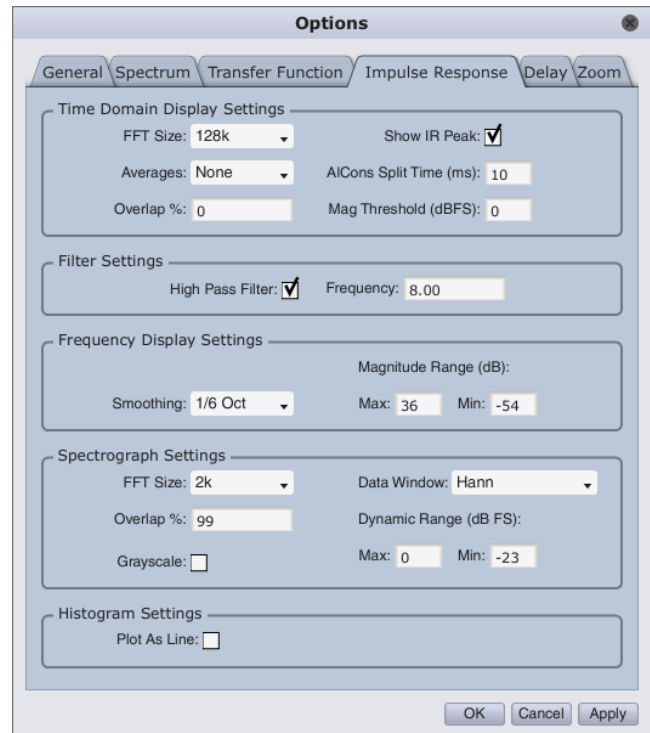
- *Use Meters/Celsius* – when this option is selected, Smart displays distances in meters and the temperature used for calculating speed of sound in degrees Celsius. Otherwise Smart displays distances in feet and uses degrees Fahrenheit for temperature.
- *Speed of sound ([unit]/sec) and temperature* – 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

- *FFT Size* and *Averages* echo the settings of the on-screen controls for dual-channel IR measurements that we talked about previously in topic #7.
- *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. 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.

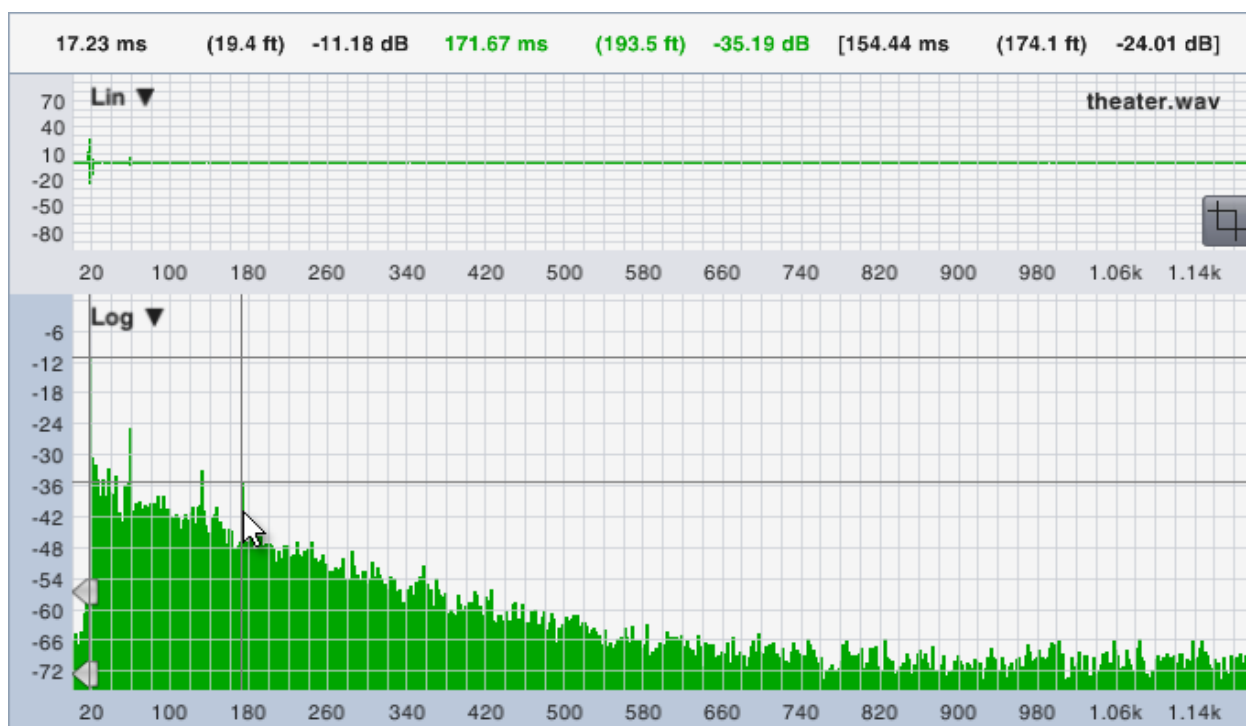


Figure 86: The logarithmic (Log) time domain IR graph plots time on the x axis and magnitude in decibels on the y axis. The combination of locked and movable cursors enables you to find time and level differences between any two points on the plot. Time coordinates can optionally be plotted with equivalent 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 <http://rationalacoustics.com/support/800456-Smaart-v7-Documentation> and select *Sample IR Wave Files* from the *Smaart v7 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.

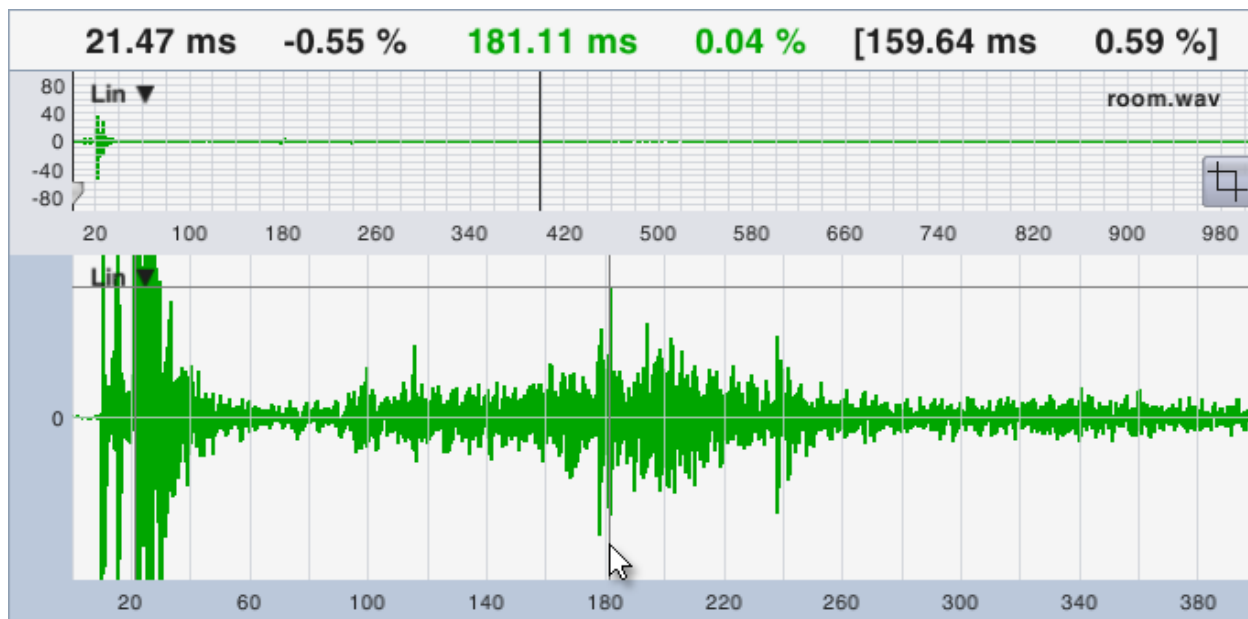


Figure 87: Zooming in on a Linear (Lin) time domain view of *room.wav* and using the cursor readout to find the relative arrival time of a prominent discrete reflection.

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.

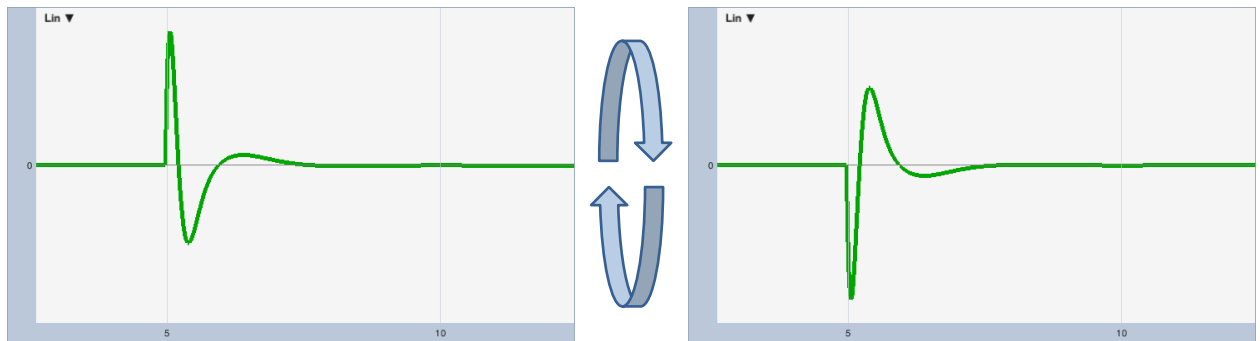


Figure 88: Zoomed in views of the linear impulse response of a bandpass filter, with normal and inverse polarity.

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 88 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.

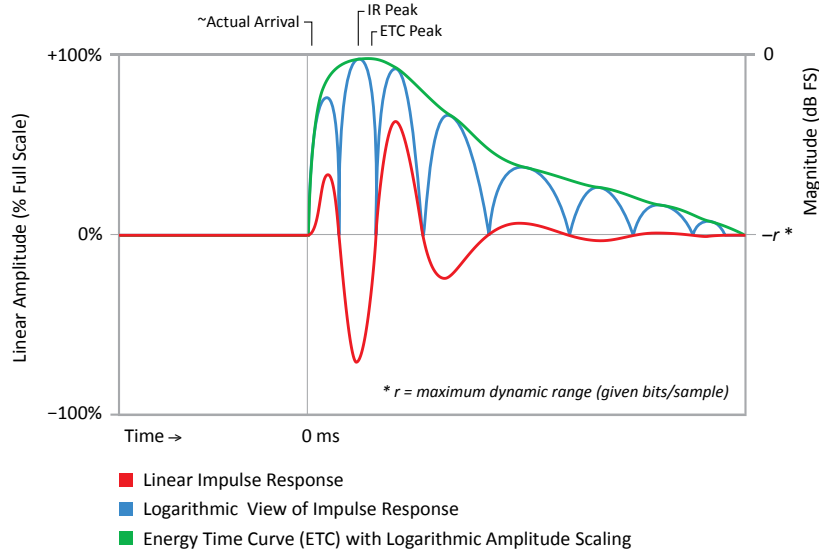


Figure 89: A comparison of the ETC and the impulse response with linear and logarithmic amplitude scaling.

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.

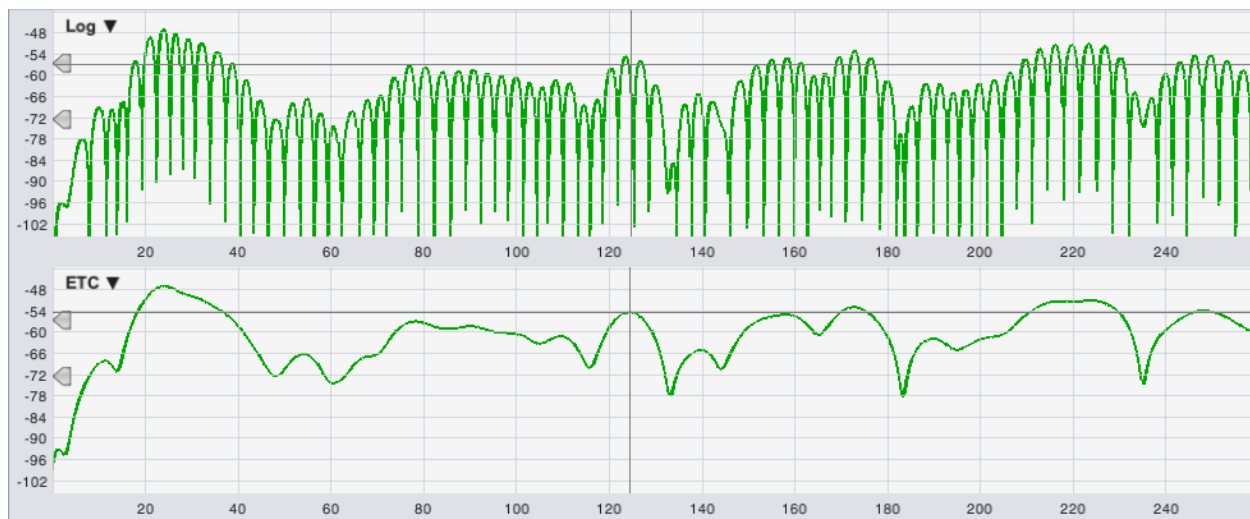


Figure 90: A comparison of the Log IR and ETC graphs in Smart for the 125 Hz octave band in room.wav.

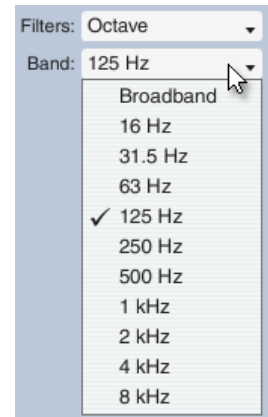
If you zoom in on the first 250 ms of room.wav and switch to the 125 Hz octave band, the difference between the Log IR and ETC is pretty striking (see Figure 90). To scale your display to look like Figure 90,

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 90, 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. At that point you're getting into the high end of the syllabic rate for very fast human speech, and people find it disorienting to be hearing the last syllable they spoke or sang while attempting to utter their next. That was actually the problem being investigated in the IR measurement shown in Figure 100 on page 132, 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. Smart 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 at the end of the file. The location of the saddle point in an IR is notoriously difficult to estimate automatically though. Smart (7.5 or higher) 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.

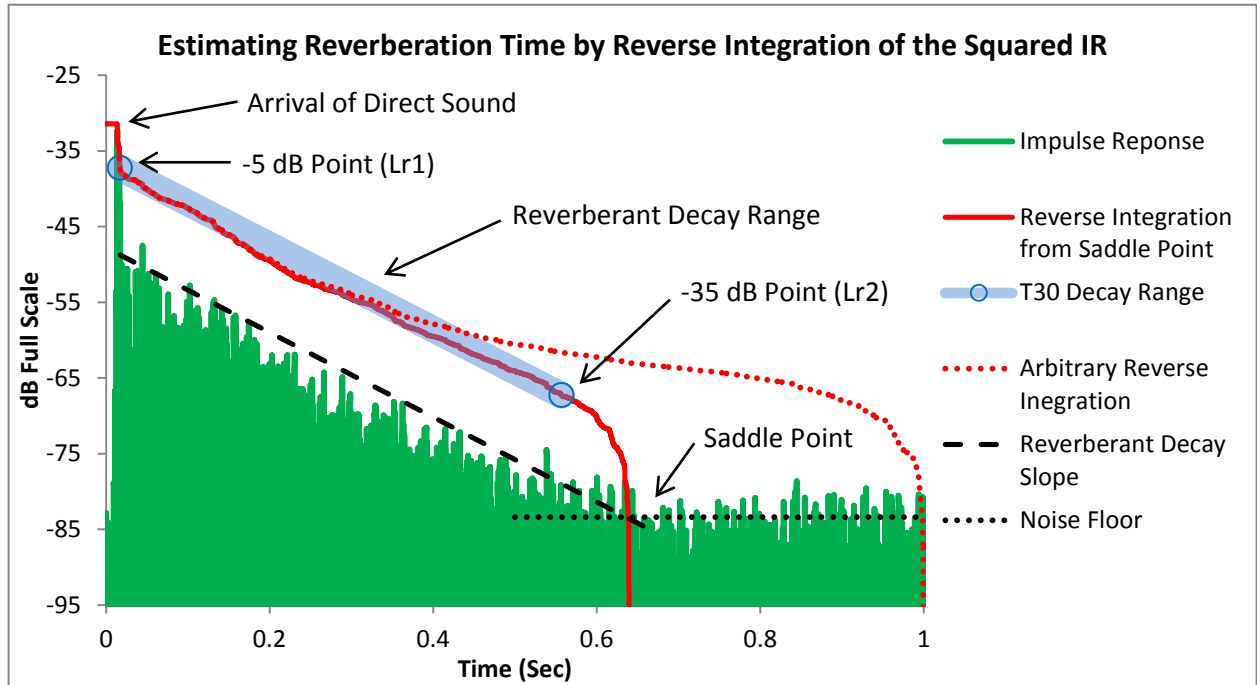


Figure 91: Estimating Reverberation time by reverse integration of the impulse response. Reverse integration of the IR from the “saddle point” – the approximate point where the reverberant decay slope meets the noise floor of the measurement – provides a very good estimation of reverberant decay time. 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 92. If you were wondering about the cryptic labels, your secret decoder and the default positions for each of the markers is as follows.

- L_d = Level Direct. This marker is positioned on the reverse integration curve at the point corresponding to the arrival time of direct sound.
- L_e = Level Early (Decay). This marker is automatically positioned 10 dB down from the L_d marker on the reverse integration curve. The slope between L_d and L_e 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. As of version 7.5, Smart 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.

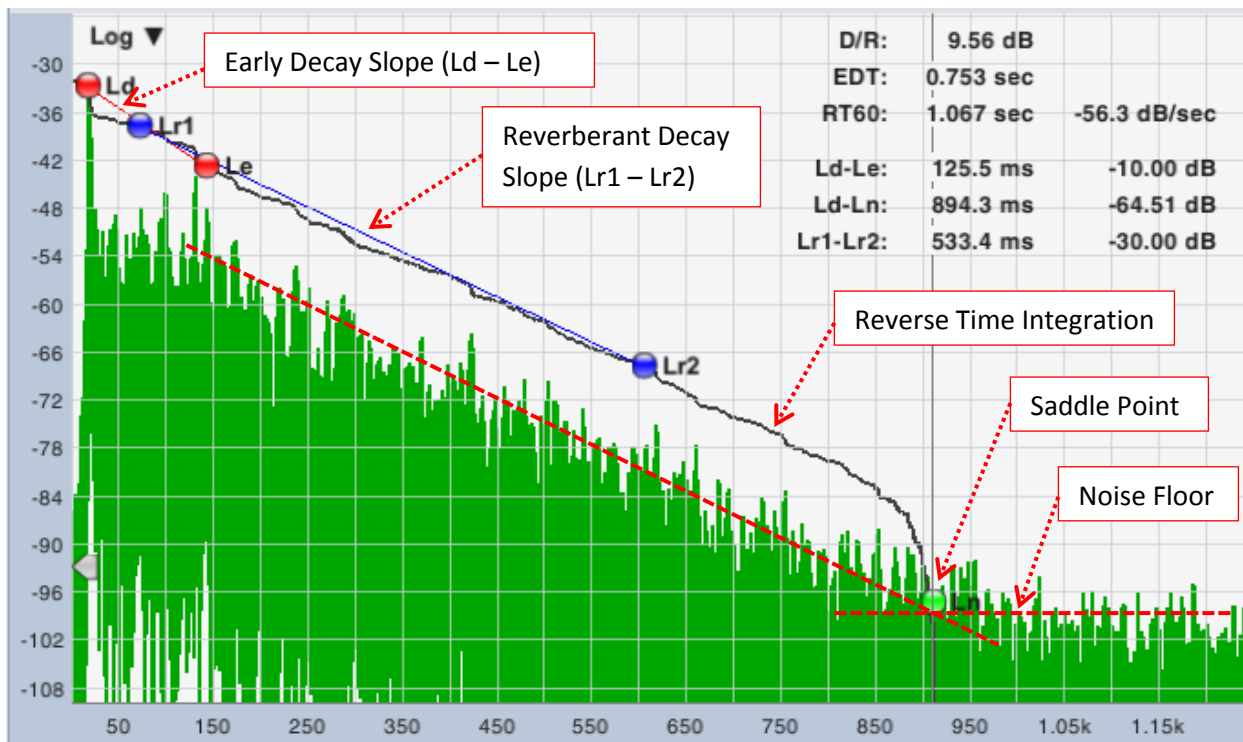


Figure 92: A log IR display with all the bells and whistles. The impulse response shown here is the 500 Hz octave band of theater.wav. Clicking the Schroeder and RT60 buttons displays the reverse time integration curve and the start and end points for the EDT and RT60 evaluation ranges on Log or ETC 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 L_d - L_e 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 L_d - L_n 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 L_e 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 the 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 TLow figure. The average of the 500 Hz and 1 kHz band is called TMid. When a single number figure is given for reverberation time, it is assumed to be TMid unless otherwise stated. Smaart calculates these values for you automatically and displays them in the All Bands Table.

Dividing TLow by TMid 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

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 Marshall used an octave higher for speech and two octaves higher for music that it 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*.

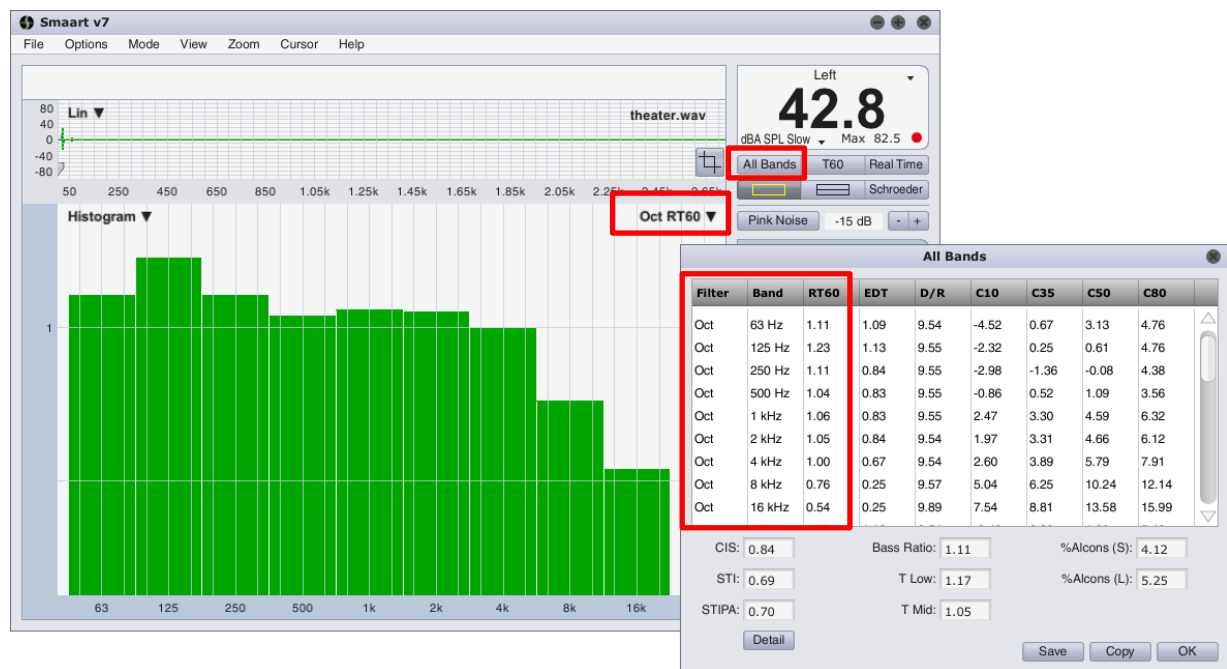


Figure 94: The Histogram graph and All Bands Table. The All Bands Table in Smaart collects reverberation times and early-to-late energy ratios for all octave and 1/3-octave bands in a single table. Speech intelligibility metrics (STI and ALCons) are displayed here as well. The Histogram chart can plot 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

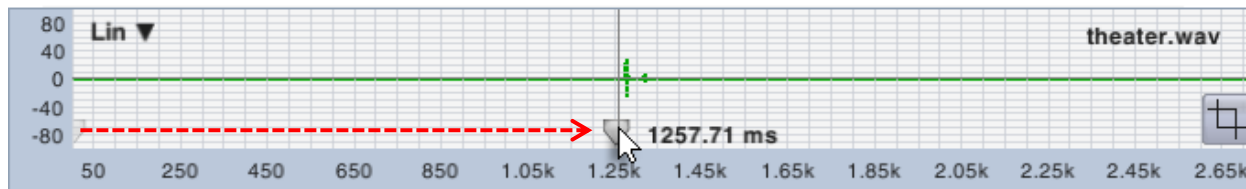
Clicking the *All Bands* button in Impulse mode in Smaart 7 brings up a report window containing just about every acoustical quantity that Smaart can calculate automatically from an IR, for each octave band and 1/3-octave band where applicable (see Figure 94). Speech intelligibility metrics (STI and 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.

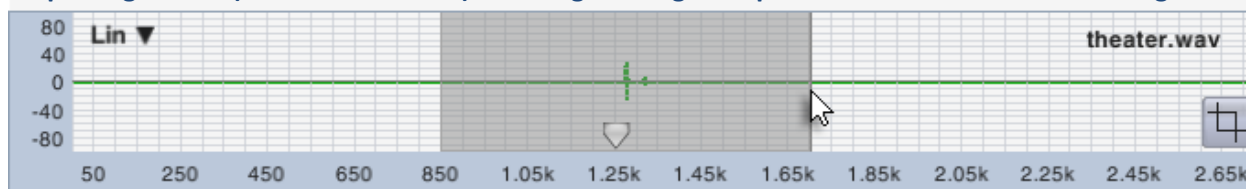
Frequency Domain Analysis

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 left 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 95: 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 Smart 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.

Smart 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 a bit, back 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 64. 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 +/- 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.

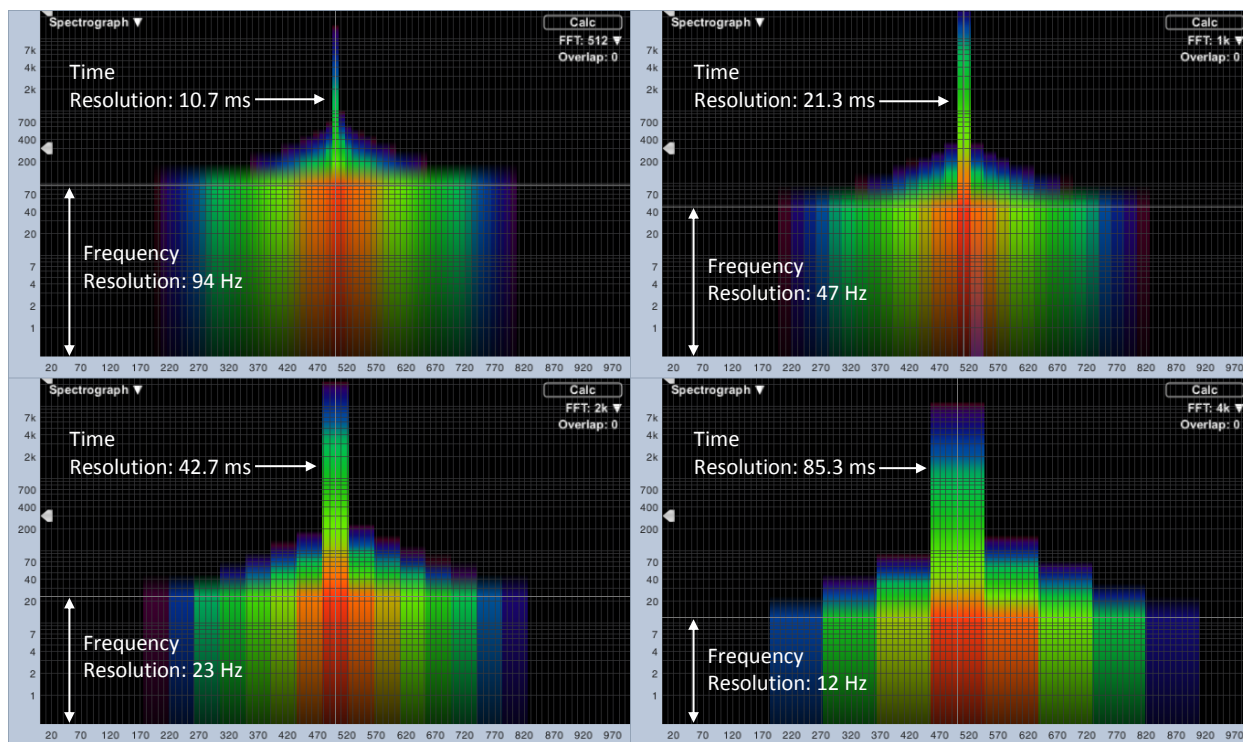


Figure 96: Spectrograph time and frequency resolution as a function of FFT size. FFT sizes ranging from 512 samples to 4K samples are compared, using 0% Overlap. As the FFT size is increased, frequency resolution improved but the peak of the IR is smeared out over a wider time range. The x axis of the graph is time, with frequency on the y axis.

Spectrograph Time and Frequency Resolution

In *Spectrograph Basics* in Chapter 5 (page 64), we used an octave band analyzer as an example for the sake of simplicity. Smart 7’s 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.

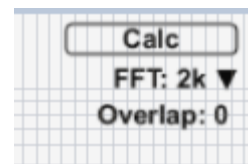


Figure 96 illustrates how this relationship works. It was created using the file 6dbOctImpulse.wav; the impulse response of a linear phase lowpass filter with 6 dB per octave roll-off. The sharpest part of the peak in the impulse response, where most of the HF energy lives, is probably no more than a 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

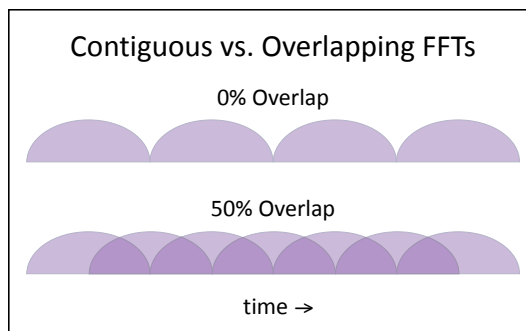


Figure 97: FFT overlap. When set to 0%, each FFT is calculated from unique data. At 50% overlap, the darker shaded areas are shared by successive FFTs. Our "FFT's" are drawn as flattened half circles to suggest a tapered FFT data window.

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 96, 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 97).

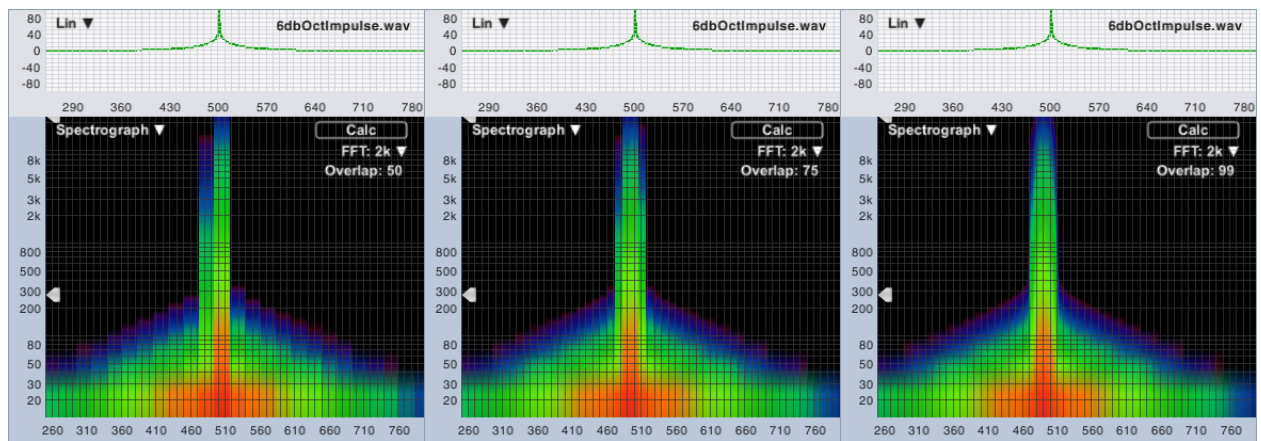


Figure 98: The 2K FFT example from Figure 96, shown with 50%, 75% and 99% overlap (left to right).

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.

Spectrograph Analysis of an Acoustical Impulse Response

Long-time Smaart users may actually recognize the impulse response shown in Figure 100. It was made using the file *room.wav* that was distributed with early versions of Smaart. A spectrograph of this file first appeared in the Smaart 1.0 user manual and it has popped up in innumerable places since then. 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

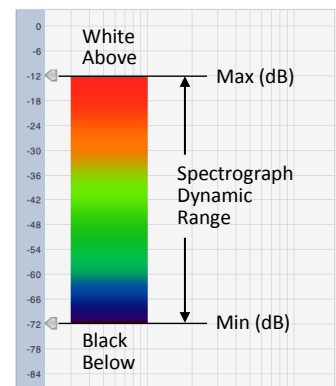


Figure 99: Spectrograph dynamic range and color mapping.

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.

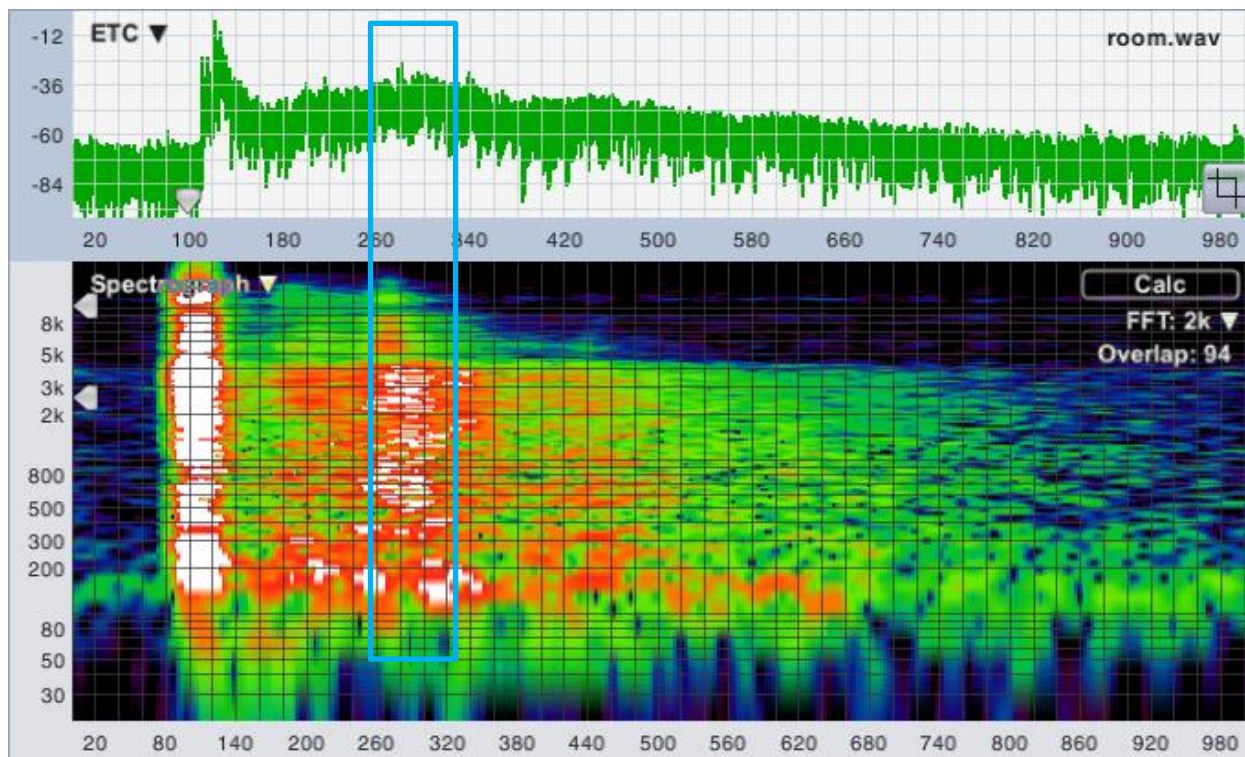


Figure 100: Broadband ETC and Spectrograph of a room impulse response showing a problematic back wall reflection. The spectrograph can be extremely useful for examining both the level and frequency content of features in the IR such as reverberant build-up and discrete reflections.

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 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 frequency (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. Looking at Figure 101, a segment of actual human speech, it's not hard to see how someone might arrive at that conclusion.

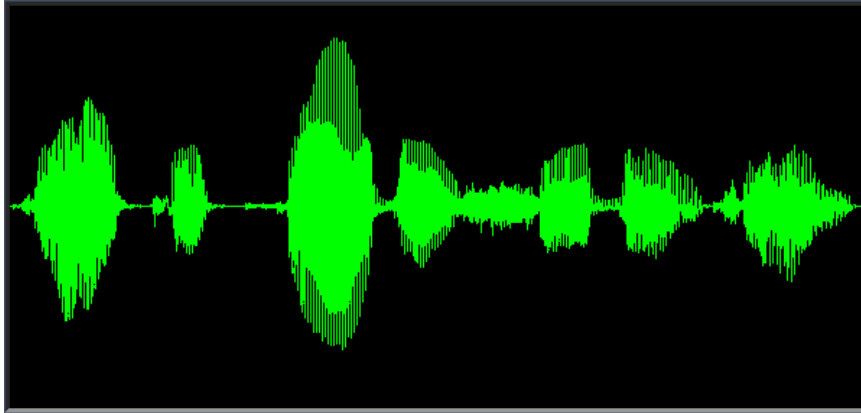


Figure 101: Recording of a male voice saying, "Joe took father's shoe bench out."

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." However to us, this seems to overlook the fact that frequency bins in a DFT are linearly spaced, meaning two lowest bins are always an octave apart, whereas the modulation frequencies for STI are on 1/3-octave intervals. If the DFT time window is 1.6 seconds then the two lowest bins are 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 5 seconds was a sweet spot, producing data points very close to all of the STI modulation frequencies. That is the reason for the 5000 ms DFT size introduced in Smaart 7.5 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.

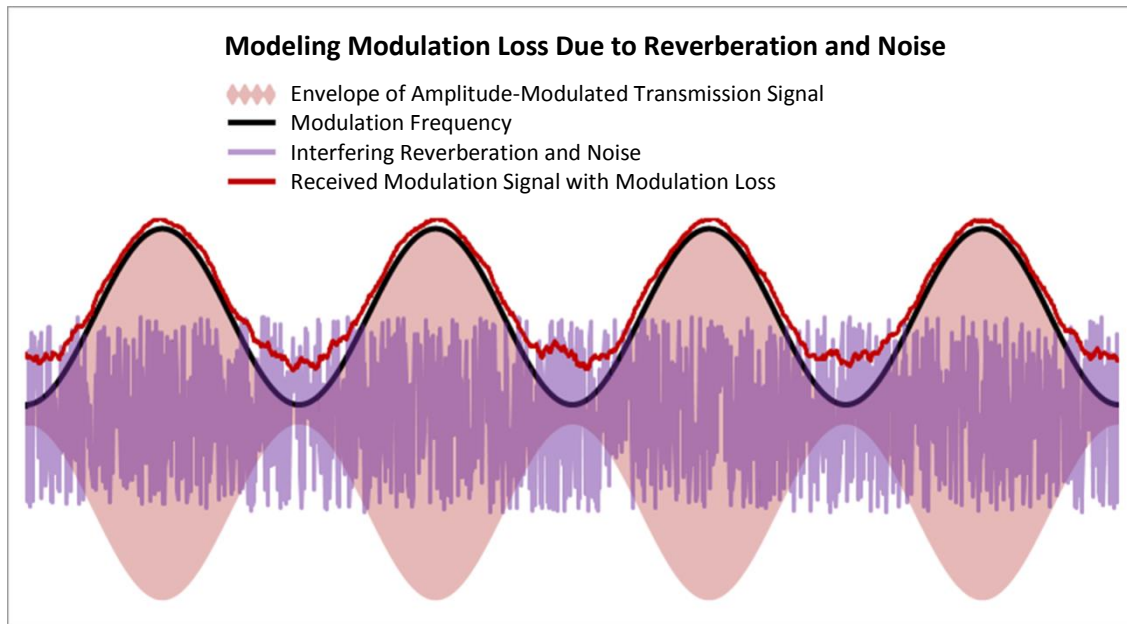


Figure 102: STI estimates speech intelligibility through a transmission channel as a function of modulation loss. The black line in the figure above is the modulation signal for a transmitted signal. The red line is the modulation signal of the received transmission. The difference between the two (the modulation transfer function or MTF), quantifies loss of modulation due to factors such as reverberation and noise.

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.

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. There are 98 modulation frequencies in all (14 x 7), and 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. Consonants are critical to speech intelligibility, as they are short in duration and tend to get lost more easily than vowel sounds that are voiced over a longer period of time.

ALCons was originally conceived as an estimate based on distance of the listener from a sound source, the directivity (Q) of the source, the volume and reverberation time of the room and the level of background noise. A longer form of the calculation suitable for direct measurement was later developed and found its way into various acoustical measurement platforms including Smaart. The long form uses direct vs reverberant energy – usually based on C20 or C10 – in place of Q, distance and volume. ALCons can be calculated for any frequency range but it is conventionally regarded as 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 form, it made estimation of 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 long form, it could be directly measured in existing installations based on ETC or IR data produced by the TDS and MLS/FHT

Chapter 9: Analyzing Impulse Response Data

measurement systems that were prevalent before the computing power needed to calculate FFTs large enough for room acoustics work became widely accessible. Its main disadvantages are its reliance on assumptions that the sound field is statistically well behaved and without audible echoes, which may or may not be true.

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 less 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 uncertainty 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 Smart makes this easy to do.

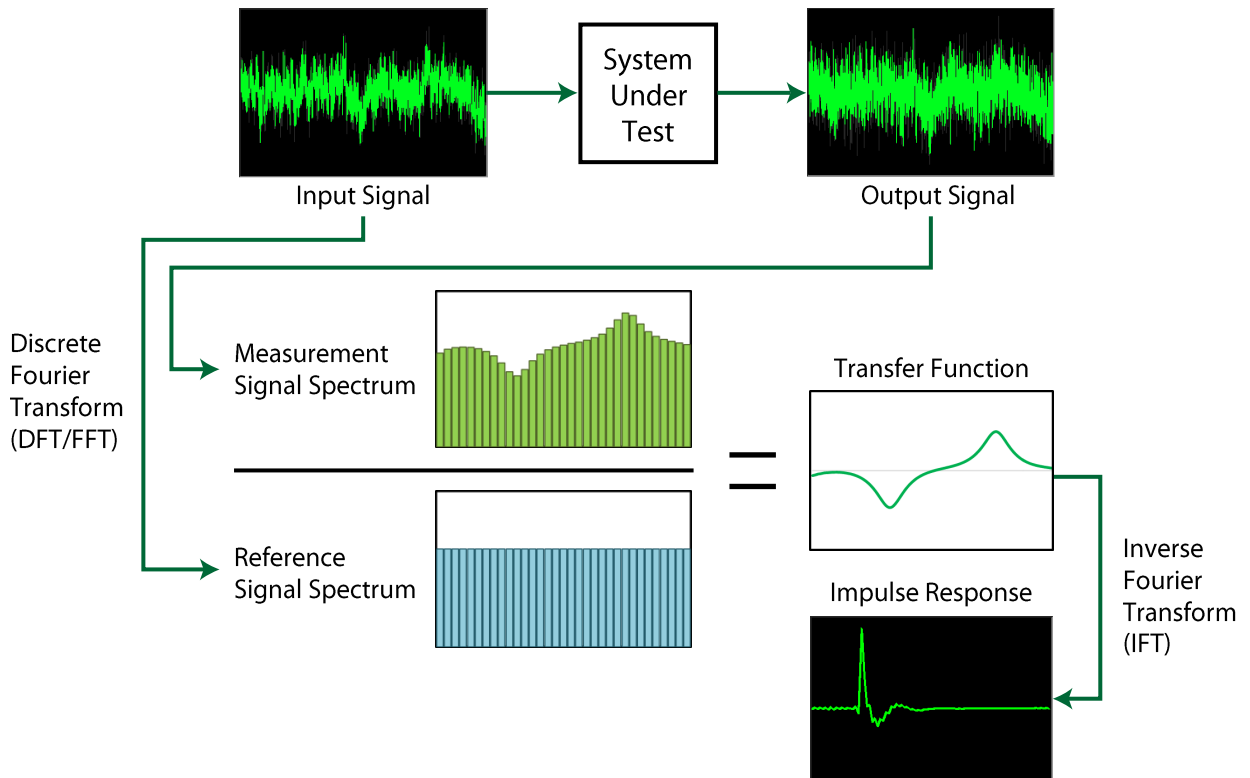


Figure 103: Block diagram of a Dual-FFT transfer function IR measurement.

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 cyclicity 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.)

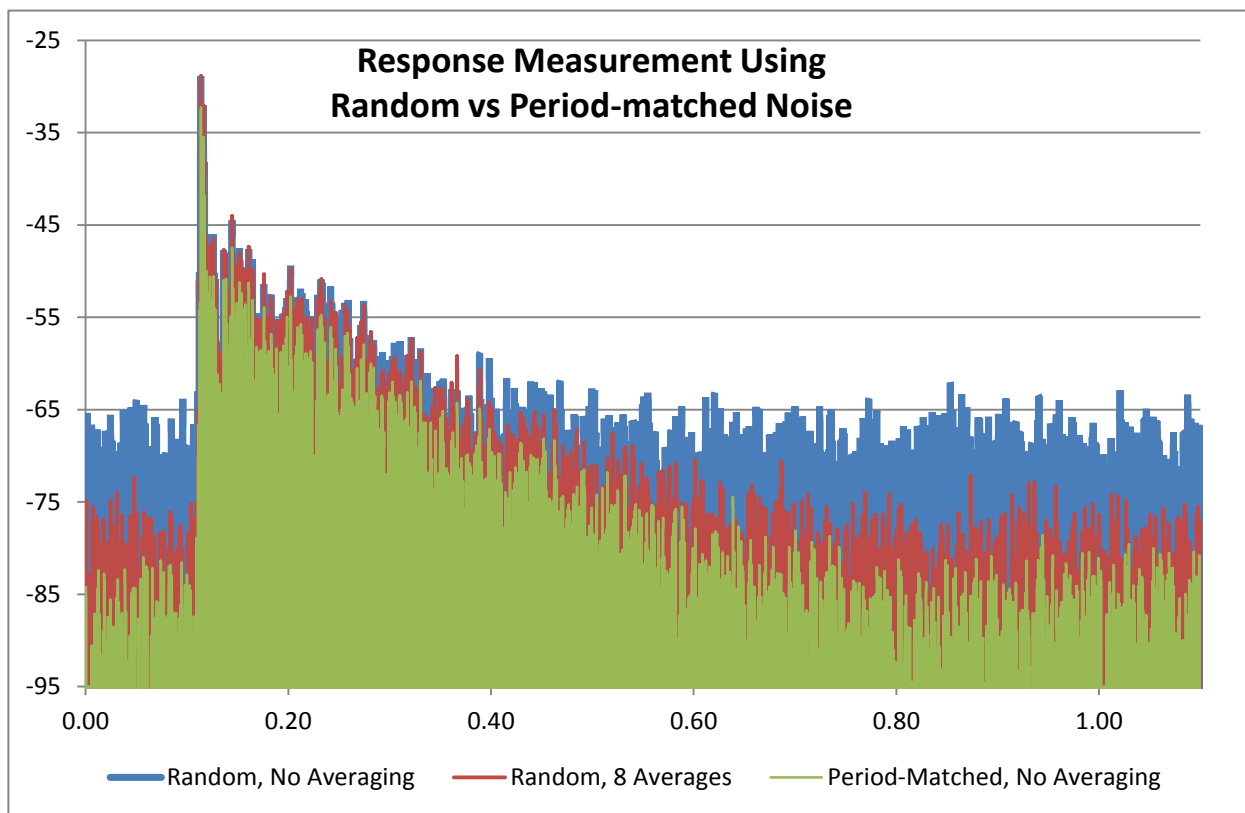


Figure 104: Three indirect IR measurements of the same room, taken from the same microphone position using effectively random noise vs period-matched pseudorandom noise. The period-matched noise measurement (in Green) takes the same amount of time as the unaveraged random measurement (in Blue) but has 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 these artifacts do not intrude on the reverberant decay slope.

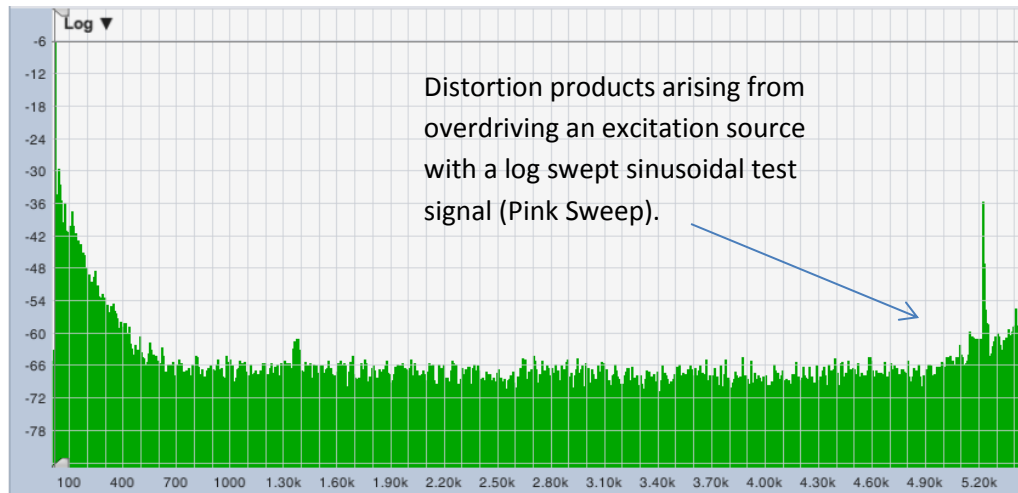


Figure 105: An impulse response measured using a log swept sine signal (Pink Sweep), showing harmonic distortion products from the excitation loudspeaker piled up at the end of the time record. In this case, the speaker being used to excite the room was overdriven and the distortion component was quite 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.

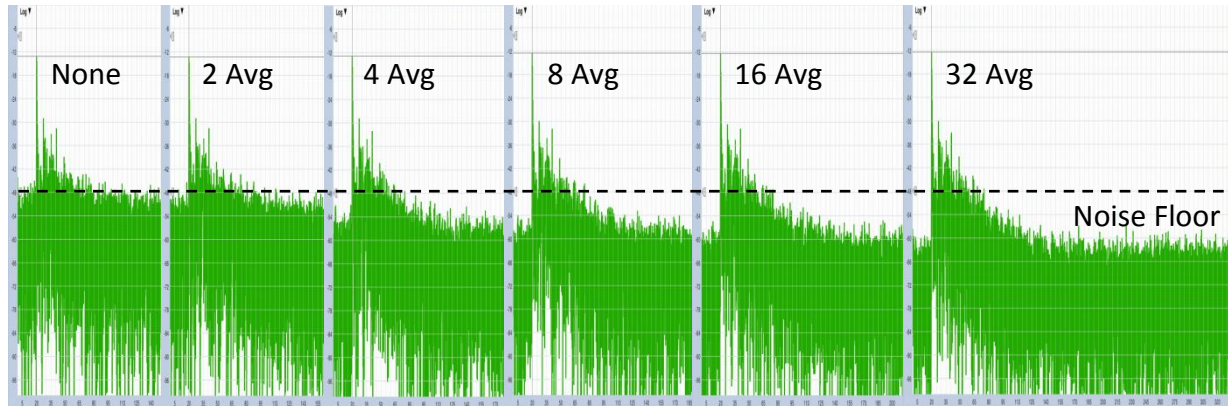


Figure 106: The effect of averaging on an IR measurement made using random stimulus signals. In theory, each doubling of the number of averages increases signal-to noise ratio by 3 dB.

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.

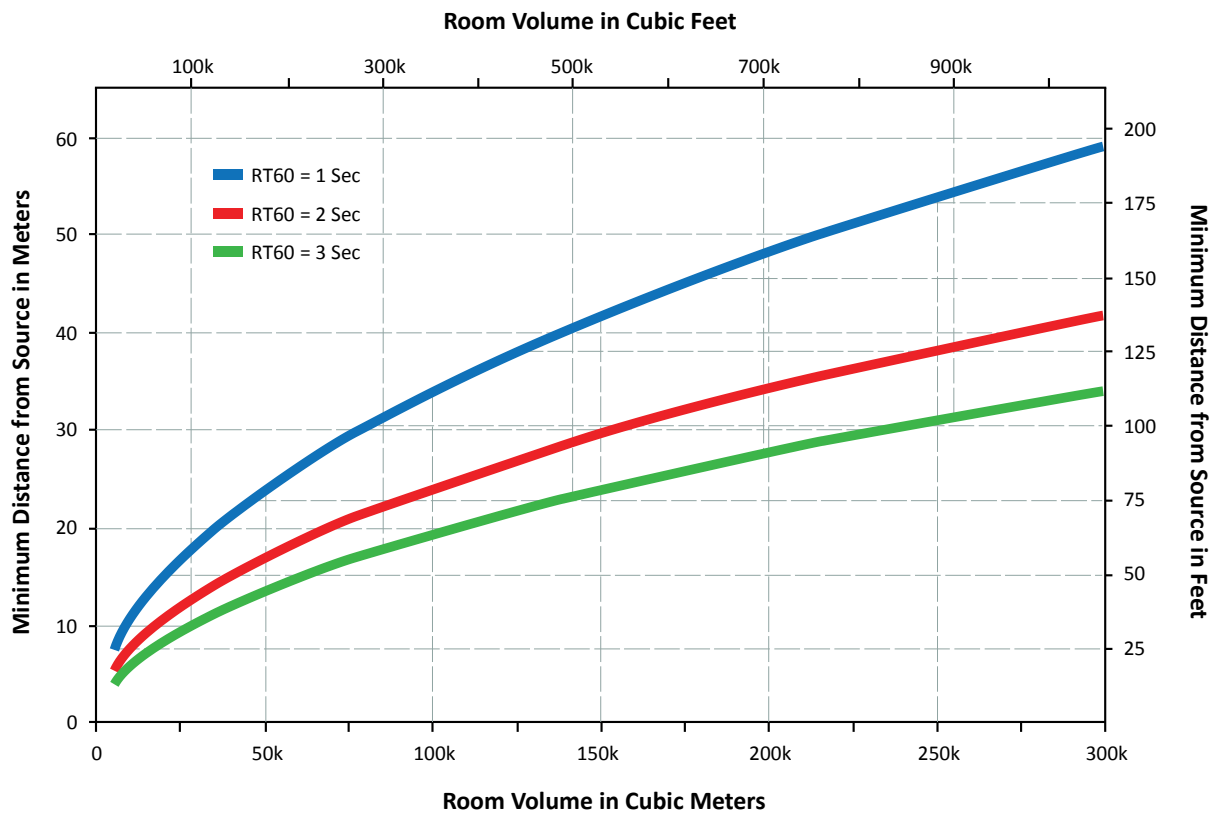


Figure 107: Minimum distance to any measurement position from the excitation source (e.g., a loudspeaker) used for reverberation time measurements. The minimum distance is a function of room volume, estimated reverberation time and the speed of sound, as described by Eq. 1 above. This example uses speed 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 107 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 $\pm 10\%$ for octave bands. The Engineering method calls for at least two stimulus source positions and six independent source-microphone combinations for a nominal accuracy $\pm 5\%$ for octave bands or $\pm 10\%$ 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 $\pm 2.5\%$ for octave bands and $\pm 5\%$ 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 108). 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 strip) 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.

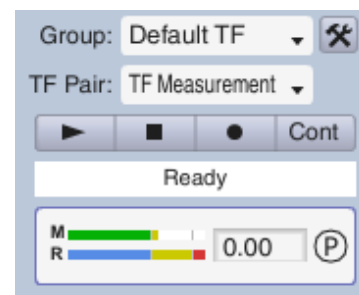


Figure 108: The measurement signal level (M) is running a comfortable level. The reference (R) channel is clipping.

Measurement Duration (Time Window)

For dual-channel IR measurements, the time window is a function of FFT size (see Figure 109). 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 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.

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, Smart 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

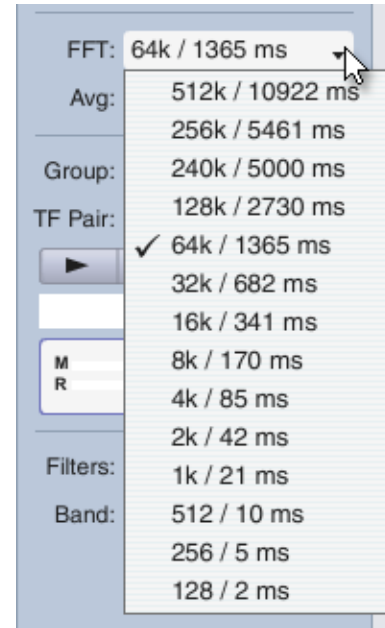


Figure 109: IR Mode FFT size selector showing the time constant in milliseconds for each

of averages. When you set the measurement *Overlap* to a non-zero value then successive FFTs share some data in common (see Figure 97 on page 130) – 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.

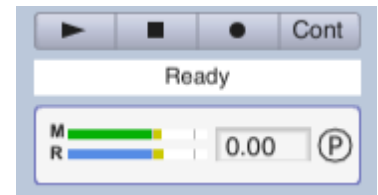
Delay Compensation

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. In Smaart 7.4 or higher, the button labeled “@” (for Peak) that appears next to the input level meters in IR mode (see Figure 108) 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 strip on the right side of the main window (see Figure 108). 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

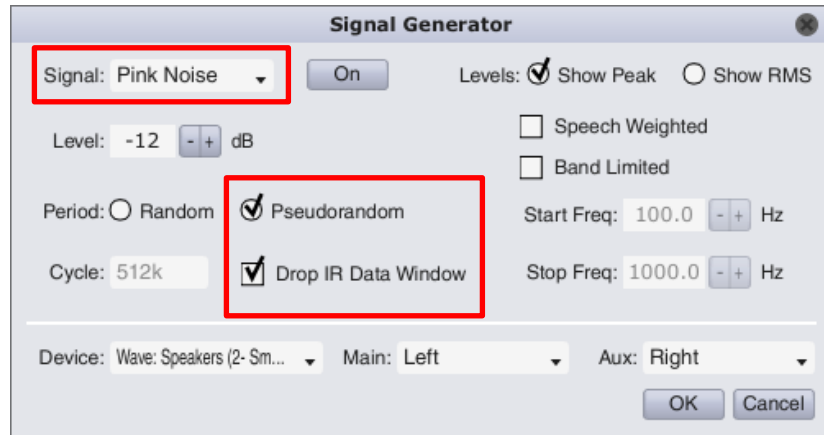
When you measure an IR in Smaart does not automatically assume that you want to save it. Sometimes it takes a few tries to get it right and we don't need a lot of old baggage piling up in the process. When you have a measurement that you are happy with and want to save, click in the File menu and select 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: Recommended Default Settings for Dual-channel IR Measurements

The following are some recommended “go-to” default settings for IR measurements in Smaart that should typically work well for most rooms.

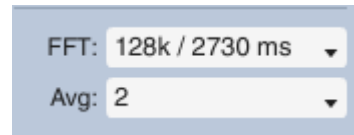
Signal Type

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, -10 to -12 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

Appendix A: Applicable Standards and Further Reading

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 taken for the 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 meters 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)

Material Absorption Coefficients

Typical absorption coefficients:

| Material | 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 |

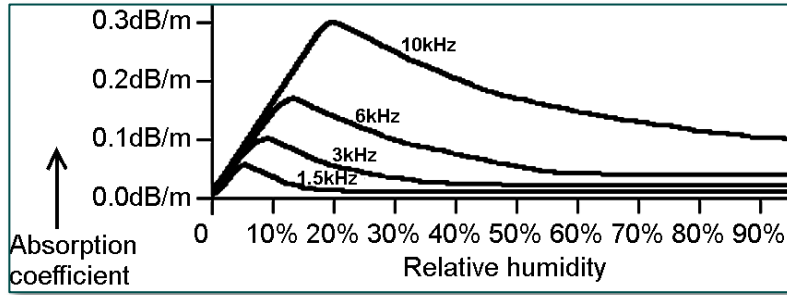
Note that absorption coefficients vary with frequency.

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.

www.doctorproaudio.com/doctor/calculadores_en.htm#calc_humid



Air absorption (in dB/m) vs RH 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

RT60 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. 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 RT analysis.

The crossover between the Schroeder region and the Modal region is known as the Schroeder frequency. This frequency only marks a cross-over region where modes start to overlap by a fairly arbitrary factor of three. It is not an exact science. A popular approximation based on Schroeder's work is:

$$f = 2000\sqrt{RT}/V$$

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:

$$f = 11885\sqrt{RT}/V$$

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 rate of decay.

Typical values are as follows: (in practice, of course, acceptable ranges vary with expectations – based on the venue size)

- | | |
|------------------------------|-----------------------|
| ▪ Talk studio | less than 0.5 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 |

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:

$$L_p = L_w + \log_{10} \left(\frac{Q}{4\pi r^2} \right) + (4/R)$$

The directivity factor (Q) can also be used to estimate the critical distance (D_c).

$$D_c = 0.14 \times \sqrt{(QR)}$$

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 troubleshoot 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)

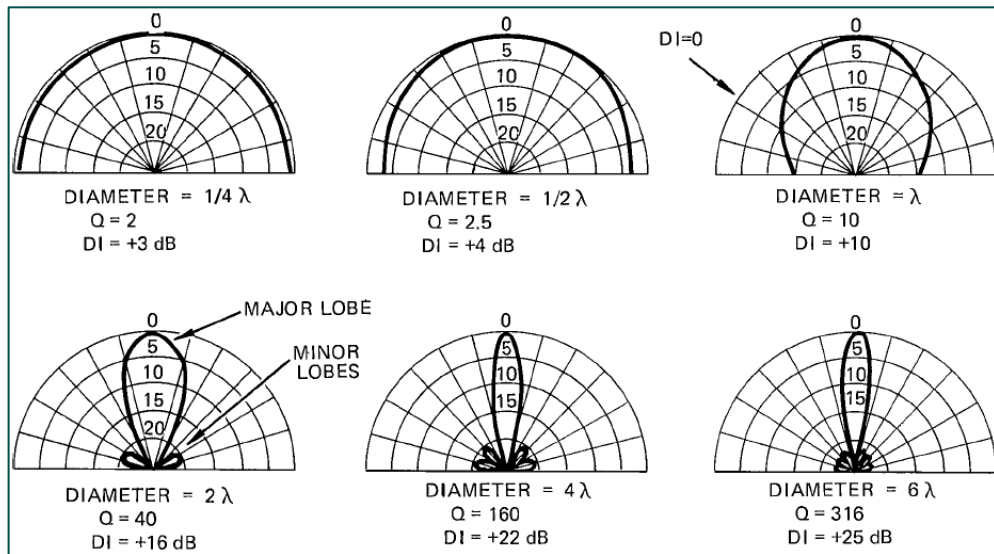
An alternative directivity figure that indicates the concentration of radiated power in dB is the Directivity Index (DI). Its value (in dB) is:

$$DI = 10 \times \log Q$$

So, assuming the listener is on axis, if $Q = 2$, $DI = +3\text{dB}$; if $Q = 4$, $DI = +6\text{dB}$ etc...

Q and DI figures in Practice

Large radiating surfaces tend to be more directional than small ones. Whilst 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.

Appendix C: Sound Source Characteristics

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 Smart to check that installed systems are meeting spec. Smart'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.

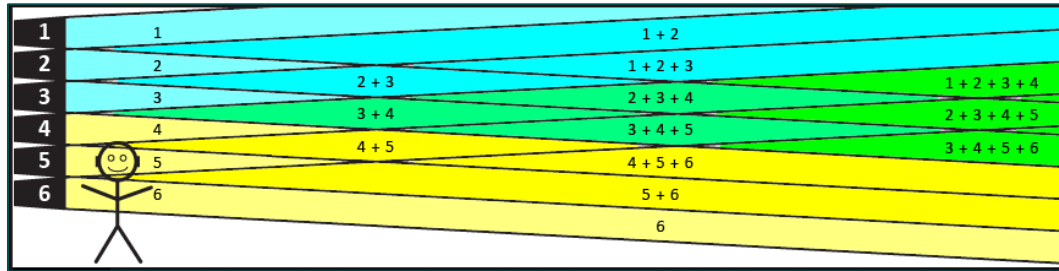
In reality, there is rarely the budget for a straight line source column as it would have to stretch from the floor to the highest seat. Most “line” arrays are much shorter for budget and safety reasons so have to be progressively curved from top to bottom in order to cover the seating areas from the 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.

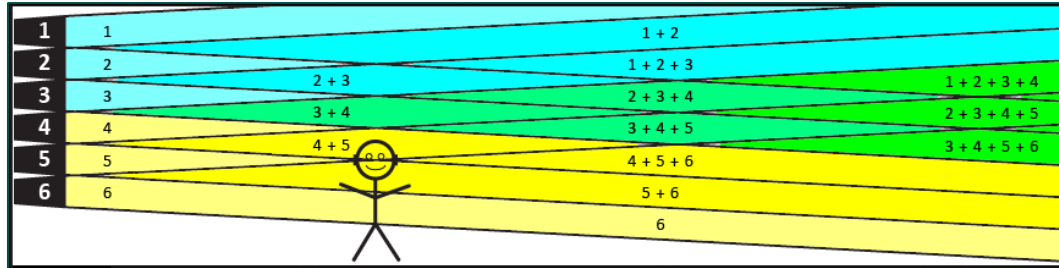
A Note about Echoes

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 Smart'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.

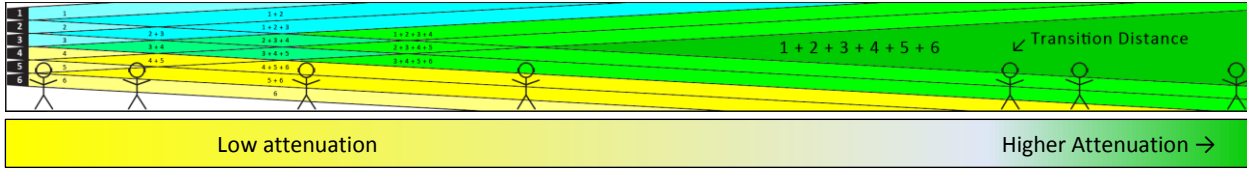


Each element (#1-6) is a point source and would, if measured in isolation, have the usual 6dB/doubling of distance radial attenuation characteristic. However, as Victor moves further away from element 5, he starts to benefit from the vector sum of elements 4 and 5 so the usual 6 dB/doubling characteristic 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.

Appendix C: Sound Source Characteristics



Note that the transition distance, for a true line source, is proportional to frequency and to the line length².

$$Transition\ Distance = \frac{Line\ length^2 \times frequency}{2 \times speed\ of\ sound}$$

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.

- Q) What has all this got to do with measuring room acoustics using Smart’s Impulse mode?
- A) Smart’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

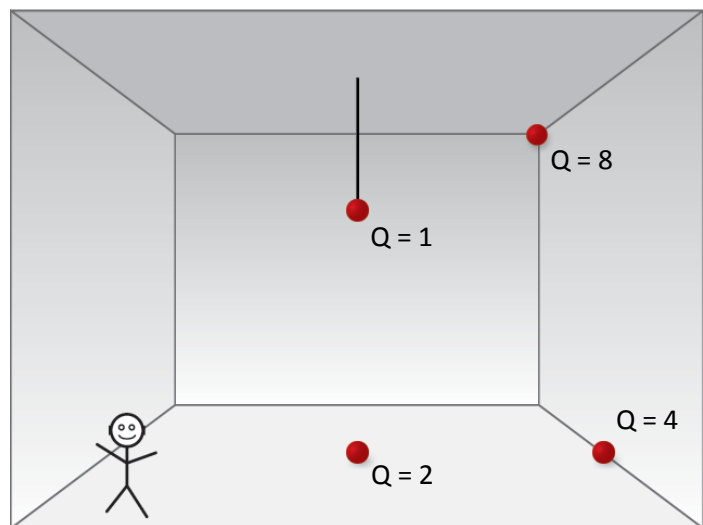
Acoustically Small Sources

Acoustically small sound sources are sources whose dimensions are small compared with the sound 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:

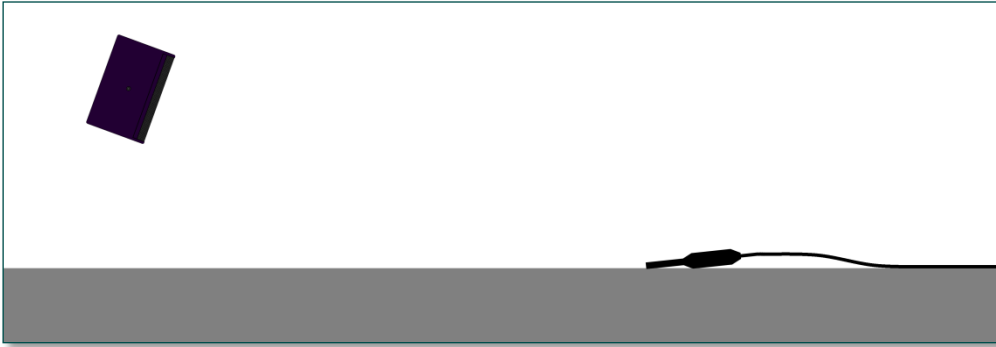
- +6dB for Q=2 (half space)
- +12dB for Q=4 (quarter space)
- +18dB for Q=8 (eighth space)

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

Appendix D: Boundary effects

devoid of seats, as the reflective qualities of the floor will only be visible as a 6dB increase in overall level.



This method has to be used with caution though, as the microphone will indicate mid-high frequency sound pressure levels some 6dB higher than would be perceived at ear height in a full audience.

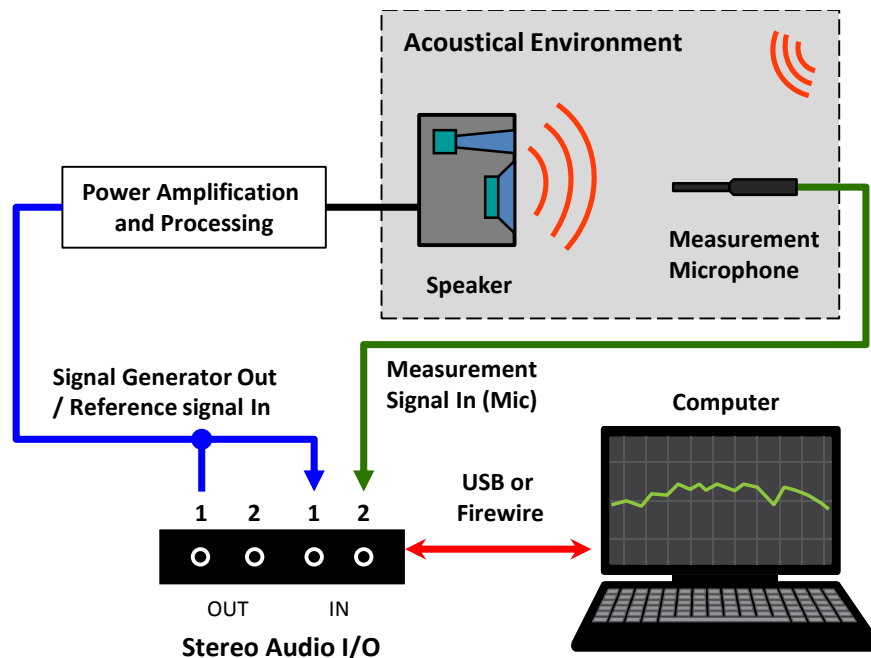
Audience members would, of course, still perceive a 6dB increase at low frequencies – because ear height is acoustically close to the floor at long wavelengths – but they'd perceive a 6dB reduction at mid and high frequencies when compared to the microphone's empty floor response.

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.



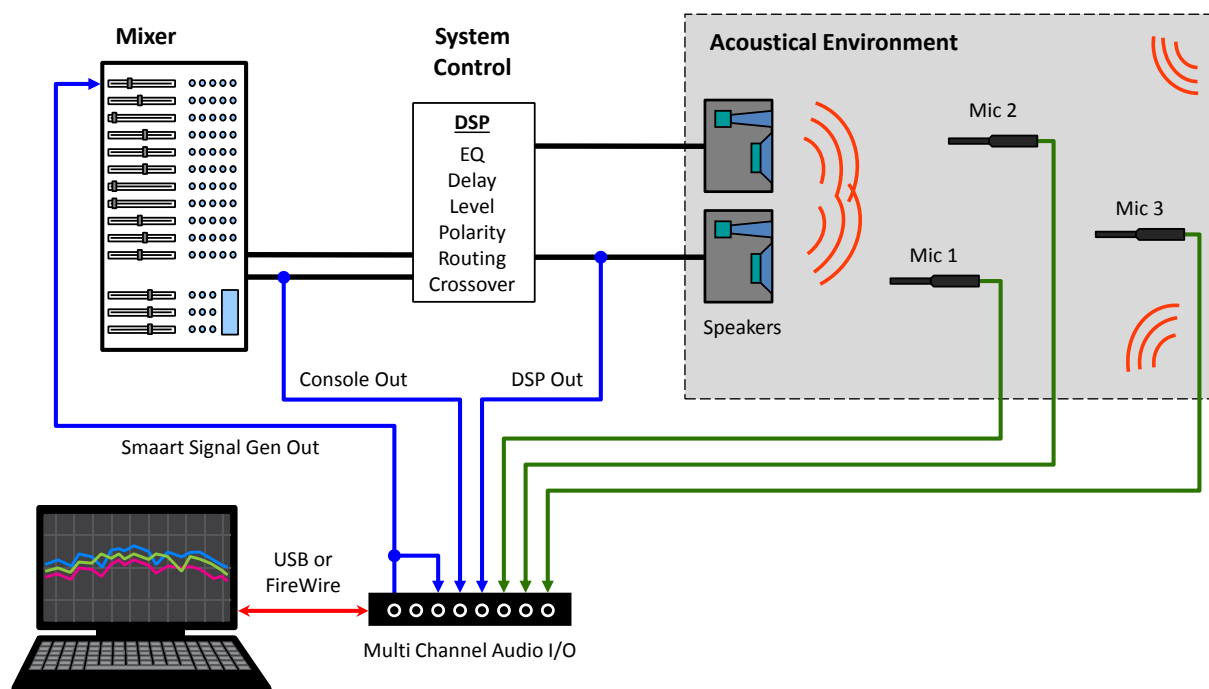
Basic connections and routing should be pretty much the same for any audio I-O device you may encounter. We trust that you understand the audio cabling and connections necessary connect your equipment together. Note that the audio I-O device in this case could conceivably be the computer’s built-in stereo 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, Smart 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

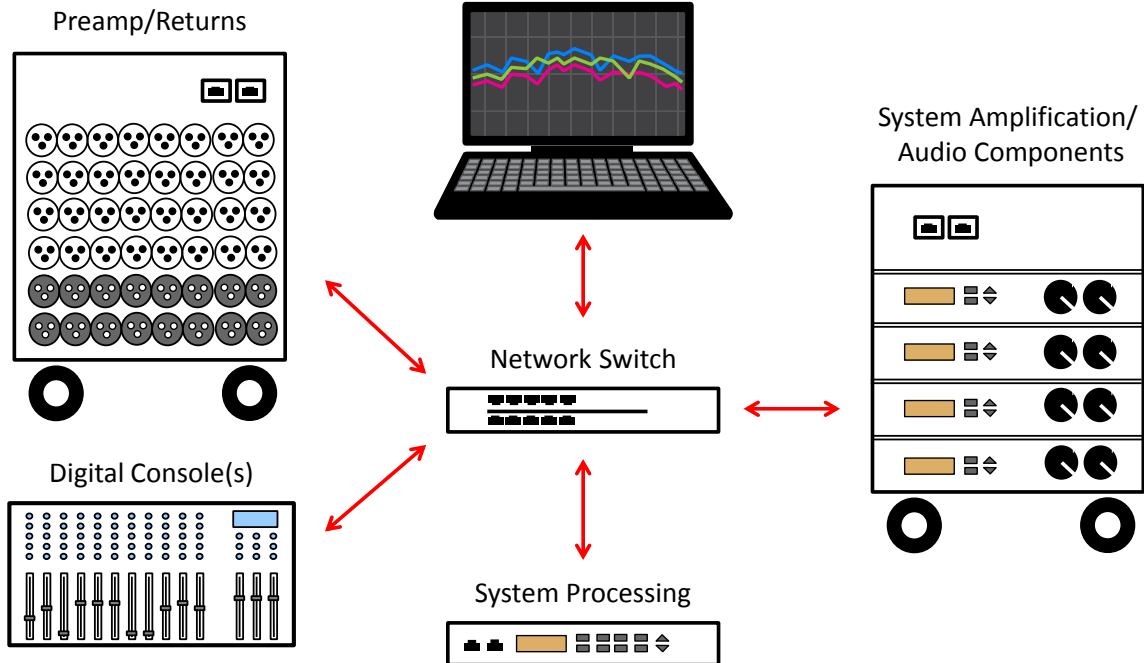
Eight-channel, single rack space packages are a common form factor for multi-channel audio interfaces. The single rack space format makes portability in a rack case, or in your backpack, equally viable. In addition to fielding more microphones, having more audio inputs to the computer enables you to 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 Smart.

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.

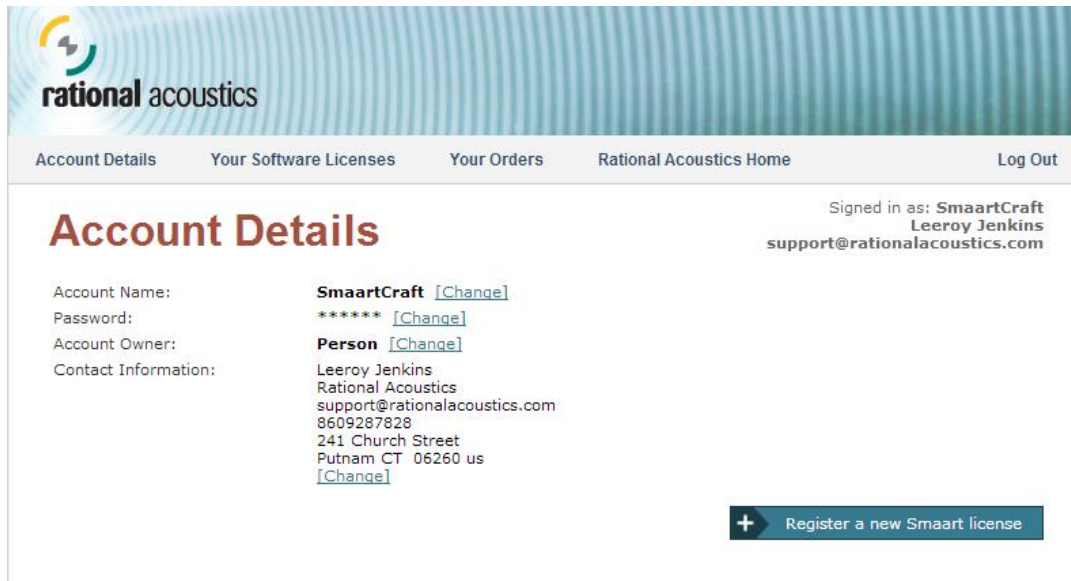
Appendix F: Licensing and Installation

The licensing system deployed with Smart v.7 provides you with greater control of your activated systems than any previous Smart version. The information in this section can help you master the license management tools in Smart 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 you purchased a new license or upgrade of Smart v.7 online, you should have received an email with a link to download the Smart v.7 installers. You can also access the latest installers at any time by logging into your user account at <http://my.rationalacoustics.com>. If you received an installation CD-ROM when you purchased Smart v.7, we still recommend visiting the web site to download the latest installer rather than installing from the CD-ROM. Installers will not appear in your licensing account until your license is registered.

If you previously purchased and registered Smart v6 you will already have a user account at my.rationalacoustics.com. Note that if you originally registered a Smart v6 license license at my.eaw.com, your Smart user account was automatically transferred to the Rational Acoustics license management site in April of 2010 and you should be able to log in using the same user name and password that you used at my.eaw.com. If you have an existing account, but have forgotten your password, the web site can send your current password to the email address associated with your account. You can also contact Rational Acoustics for assistance.



License management accounts at my.RationalAcoustics.com are not synchronized with Rational Store accounts. You must have a license management account to register your Smart license(s).

Registering a License

After purchasing Smart v.7, the first thing you should do is register the license code to an account at my.RationalAcoustics.com. License registration can be done either from within your account or through the registration dialog in the installer. If you don't have an account, you can create one from within the activation wizard -or- open your web browser, navigate to my.RationalAcoustics.com, and click on the link that reads "Click here to create a new account".



Once logged in, click the link to *Register a new Smart License* and enter your license code when prompted. The license code must be entered exactly as it appears on your package or in the confirmation e-mail for an online purchase, with dashes and full capitalization. Once a license has been registered you can simply click on it in the list of licenses on the *Your Software License* section of your user account to manage your installations.

Installing Smart v7

The Smart 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 Smart v.7 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 Vista and Windows 7 systems, administrator authority is required to perform the installation. If the login account you are installing under is not an administrator account, you will be required to supply administrator credentials during 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 Smart on your desktop.

Software Installation on Mac OS X®

Smart v.7 for Mac OS X is supplied in the form of a Mac application bundle, so installation is simply a matter of dragging the Smart v.7 icon into your Applications folder. Note that Smart v.7 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. Installing Smart v.7 signifies your acceptance of the terms of this license. Once you have read and agreed to the EULA you can drag the Smart v.7 icon in your applications folder, and then if you like, drag it from your applications folder to your dock to create a shortcut to the program.

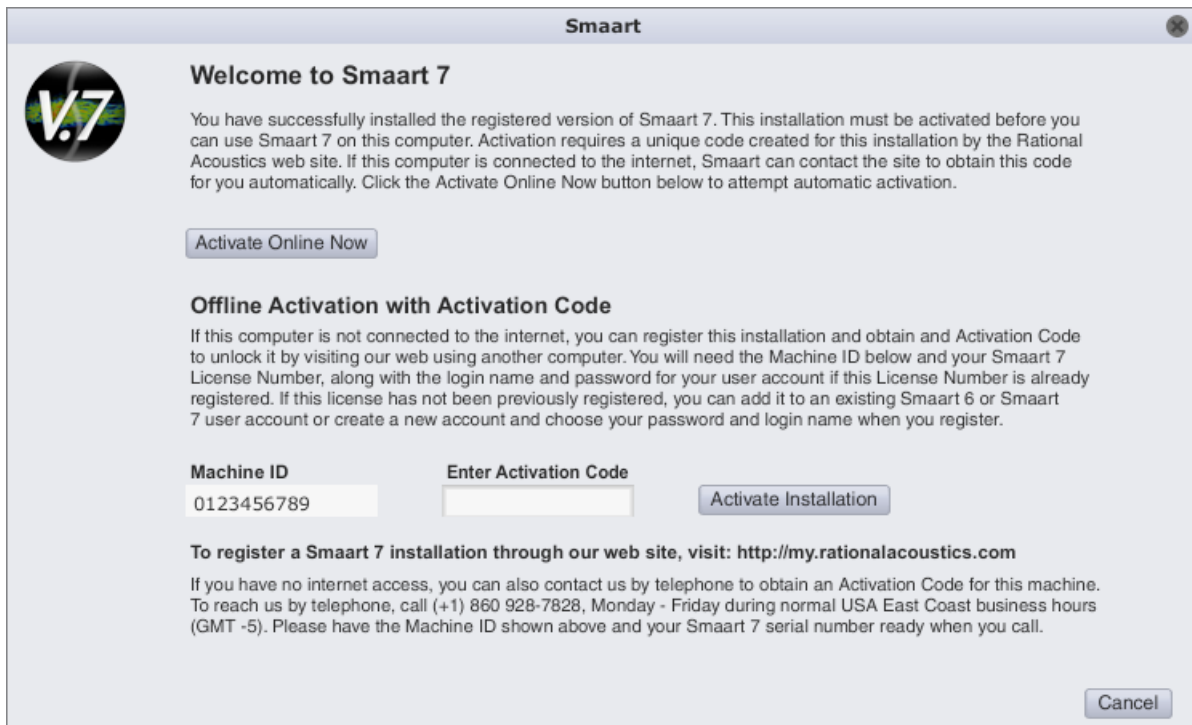
Activating an Installation

When you run Smaart for the first time, an activation screen (see below) will appear on startup. You will need to activate your installation before you can use the software.

Before you can Activate!

There are four basic requirements for activating a Smaart v.7 installation:

- You must have a valid Smaart v.7 license code (in the format: XXXXX-XXXXX-XXXXX-XXXXX).
- Your license code must be registered to a user account at my.RationalAcoustics.com. If you have not previously registered your license, you can register it at the time of installation, either through the licenst management web site at my.rationalacoustics.com or by using the online activation option in Smaart.
- You must have one or more installation spots available on your license.
- You must have internet access on or near the computer you are trying to activate.



***(Please note that the machine ID shown above is for demonstration purposes only.
Your unique machine ID will be generated by Smaart.)***

Option 1: If this computer is connected to the Internet...

If the computer you are installing Smaart on is connected to the Internet, the online activation wizard in Smaart can handle most of the details of registering a new installation for you, including creating a user account for you if you don't already have one, registering your license if necessary, and activating your installation. To begin the online activation process, click *Activate Online Now* button. This will take you to the screen shown on the next page.

At this point the process differs somewhat depending on whether or not you have previously created a licensing account and registered your Smaart v.7 license to it. You will begin by selecting the appropriate option in the left-hand column.

- ***This license has already been registered.*** Select this option if your Smaart v.7 license has already been registered to your user account at my.rationalacoustics.com. In this case all you need to know is your Smaart v.7 license code and a password for your user account.
- ***I want to register this license to my existing Smaart 6 or Smaart 7 user account.*** Select this option if your Smaart v.7 license has not been previously registered but you already have an existing user account at my.rationalacoustics.com.
- ***I have not yet registered this license and I do not have an existing online user account.*** Select this option if your Smaart v.7 license has not been previously registered and you don't have an existing user account. You can choose any login name and password for your new account.

Regardless of which option you select in the left column, you will also need to supply the following three pieces of information for each new Smaart v.7 installation in the right-hand column of the online activation screen.

- The name of the person or group who will be the primary user(s) of this computer.
- A contact e-mail address for the primary user of this computer.
- A friendly name for this machine to help you identify it when you log in to our web site to manage your installations.

This information will be stored in your account at my.rationalacoustics.com to help you tell one machine from another. It will not be used for marketing purposes or shared with any third party for any reason. When you finish filling in the required information, click the Continue button.

Appendix F: Licensing and Installation

If you are registering your Smart v.7 license for the first time, you will be taken to a contact info screen to collect or confirm contact information for your user account. If you are registering your Smart v.7 license to an existing user account, you can edit any of the information shown on the contact information screen before sending it to the server. You can even change the login name for your account if need be. Our software registration web site no longer requires that you use an e-mail address as your login name.

A Note on Managing Multi-User Licenses

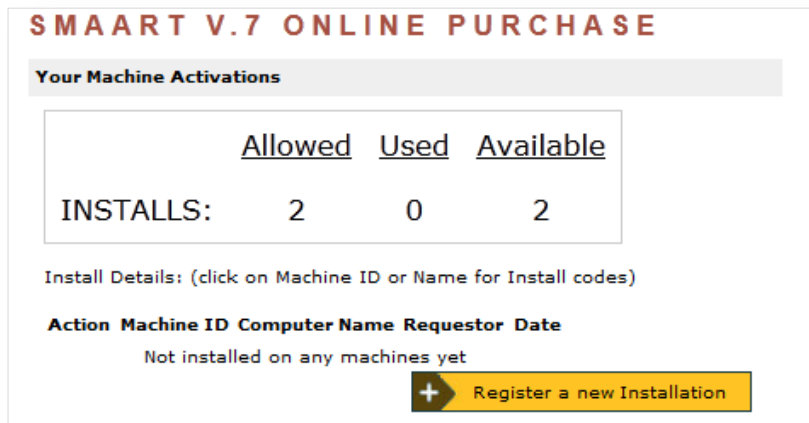
Once your account is set up, activating Smart via the online method only requires your license code and the password for your user account but does not require your login name. Organizations with multi-user licenses can therefore allow users to perform their own software installations without exposing all credentials required to administer their user account. Remember that anyone who has your license code and password can potentially activate a Smart 7 installation under your license, so be sure to take appropriate precautions to prevent unauthorized use.

Option 2: If this computer is not connected to the Internet...

If you need to activate Smart on a computer that is not connected to the Internet, you can manually register the installation from within your user account at my.rationalacoustics.com. You will need to know the computer's Machine ID, which is shown on the first screen of the activation wizard in Smart (see screen shot on page 168).

From any computer or mobile 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 all of your registered licenses. If your license is not listed on the *Your Software Licenses* page, please refer to the previous topic on *Registering a License* on page 167 for instructions on how to register it.



The screenshot displays the 'SMART V.7 ONLINE PURCHASE' interface. At the top, it says 'Your Machine Activations'. Below this is a table with three columns: 'Allowed', 'Used', and 'Available'. The row for 'INSTALLS:' shows 2 allowed, 0 used, and 2 available. Below the table, there is a link for 'Install Details: (click on Machine ID or Name for Install codes)'. At the bottom, there is a table header with columns: 'Action', 'Machine ID', 'Computer Name', 'Requestor', and 'Date'. The content below the header says 'Not installed on any machines yet'. A yellow button with a plus sign and the text 'Register a new Installation' is located at the bottom right of the screenshot.

| | <u>Allowed</u> | <u>Used</u> | <u>Available</u> |
|-----------|----------------|-------------|------------------|
| INSTALLS: | 2 | 0 | 2 |

Install Details: (click on Machine ID or Name for Install codes)

| Action | Machine ID | Computer Name | Requestor | Date |
|-----------------------------------|------------|---------------|-----------|------|
| Not installed on any machines yet | | | | |

[+ Register a new Installation](#)

Clicking on your Smart v.7 license number on the *Your Software Licenses* page takes you to a page where you can see the total number of installations allowed on your license, the number of installations you have used to date, and how many are still available. If you have no installs remaining in your

installation allowance then you will need to either deactivate one of your previous installations (see below) or contact Rational Acoustics to purchase additional installation spots for your license.

Assuming that you have a least one installation spot available, click the link to *Register a new Installation*. This will take you to a page that asks you for your Machine ID. You'll also need to provide the name and e-mail address for the primary user, and a friendly name to identify the computer in your list of installations. There is also a field for a *Block Code*, which you can ignore unless you are reactivating an installation on a computer that has previously been deactivated (see *Reactivation* on page 173 for more information).

When you finish entering the required information, click the Submit button to get your Activation Code. Once you have your Activation Code, go back to your offline machine, enter it in the field provided on the first screen of the activation wizard in Smaart (see screen shot on page 168), then click the *Activate Installation* button. That's it!

Deactivation

Smaart v.7'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 with regard to Smaart's licensing to make the process go as smoothly as possible, please see the topic on *Using Time Machine*, *Migration Utilities*, or *Cloning Software* beginning on page 172. If you are upgrading your operating system or making significant hardware upgrades to your computer, please read the topic on *Upgrading your Operating System or Major System Components* on page 173.

Moving Smaart to a New Computer -or- Upgrading 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.

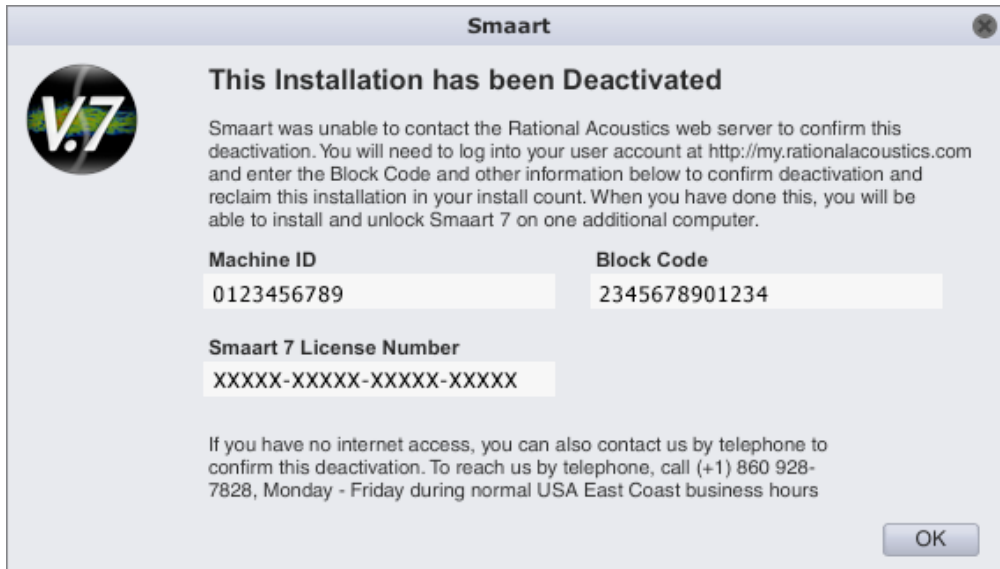
If you are completely reformatting your computer, you must deactivate Smaart first to regain your installation spot. If you plan to use Time Machine, migration utilities, or any kind of cloning software to move your file system to a new machine (or new hard drive), please see the appropriate section(s) noted above under *Deactivation*.

To deactivate a Smaart v.7 installation go to the *About* dialog window for Smaart and click the *Deactivate Installation* button. On a computer running Windows, the *About* dialog is accessible from the *Help* menu. On a Macintosh computer, select *About* from the *File* menu.

After clicking the *Deactivate Installation* button, you will be asked to confirm the deactivation. Smaart will then automatically attempt to register the deactivation with our server at my.rationalacoustics.com. If the attempt was



successful you will get a confirmation message telling you that your deactivation was received and accepted. If not, you will be presented with a screen similar to the one on the below.



If Smaart was unable to connect to the web site, 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 landing page.
2. Click on your Smaart v.7 license 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. You should see your available installations increase by 1 following a successful deactivation.

Note that your Machine ID and Block Code are also displayed on the activation screen that comes up if you attempt to run Smaart again after deactivation. If you inadvertently closed the screen above without recording those numbers or write down one of them incorrectly, you can always get back to them. *We recommend that you do not uninstall Smaart on the deactivated machine until you have confirmed the deactivation.*

Using 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 back-up 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 topic below on *Reactivating after restoring from a backup or after migrating system files*.

Upgrading your Operating System or Major System Components

If you plan to upgrade your computer's existing operating system but you are keeping the file system intact (i.e., you do not plan on reformatting the hard drive), or if you are replacing major system components such as 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 a Smaart Installation on a Deactivated Computer

When 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 field.

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 for your installation. Once you've submitted the necessary information, you will receive a new Activation Code which will reactivate Smaart.

Reactivating after restoring from a backup or after migrating system files

If you are activating an installation that's been restored from a back-up 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. If you migrated Smaart from another computer and didn't follow the instructions above (eg. you deactivated before migrating) DO NOT delete the ticket file, instead please contact Rational Acoustics.

The location of the Smaart v.7 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 deleting and installation Ticket, start Smaart. If you are asked to restore a missing file, click **No**, and when the Activation window appears, try activating again.

Mac OSX: [hard drive]/Users/Shared/Ticket

Windows Vista, 7, 8 or 10: C:\Users\Public\Ticket

Windows XP: C:\Documents and Settings\All Users\Application Data\Ticket

Note to Windows XP users: On Windows XP, the default location for the ticket file is a hidden folder. The easiest way to get to it is to click the Windows *Start* button and select *Run* from the Start menu (the

name of the *Run* utility may translate to something more like “Execute” in some language localizations), then type or paste the text below (including the quotation marks) into the *Run* dialog and click *OK*:

%allusersprofile%\application data

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.7 license. This license is registered to your account at my.rationalacoustics.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.7 license page at my.rationalacoustics.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 7 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 39 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.

Importing Spectrum Data Traces

When importing a new spectrum data trace from a text file (by selecting *More > Import ASCII* in the spectrum data bar), you are 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 automatically interpolate any missing frequency bins from whatever coordinates you supply.

Unfortunately, this means that if you try to import fractional-octave data as a spectrum trace, the resulting curve in Smaart will slope upward on a banded spectrum display at a rate of 3 dB per octave, relative to the original. It is technically possible to pre-bias fractional octave magnitude values so that the imported data curve sums to the proper fractional values, but hand tuning is typically required and the resulting curve may not work perfectly for every fractional octave display setting.

If the object of the exercise is to get a target reference curve on the screen, 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 2, beginning on page 39.

Importing Transfer Function Data Traces

When importing transfer function data traces from an ASCII file, 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 in this case is that

Smaart 7 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

A

Absorption Coefficients, 153
Accelerator Keys (Hot Keys) and Mouse Clicks, 2
Acoustically Small Sources, 161
Activating an Installation, 168
Activation, 174
Activation Code, 3, 171, 173, 174
Active Graph Pane, 25
Active Measurement, 59
Active Measurement Controls, 57
Advanced Signal Selection, 100
Air Absorption, 153
ALCons, 135
 Qualitative Thresholds, 136
All Bands Table, 127
Analyzing Impulse Response Data, 117
Anatomy of an acoustical impulse response, 106
API, 29
Application Examples
 Distortion and Overload, 69
 Examining Interaction Patterns with Spectrograph, 74
 Feedback Frequency Identification, 73
 Setting an Equalizer for a Loudspeaker, 101
Arrival of Direct Sound, 107
ASCII Export, 36
ASCII Import, 36, 176
ASCII Text File Formats, 176
Audio Device Options, 51
Audio Input and Output Devices, 50
Audio I-O, 163
Averaging, 14
 Over Time (Temporal Averaging), 15

 Spatial, 16
Averaging and Banding Controls, 61
Averaging and Overlap, 148
Averaging Stored Traces, 33
B
Band limited noise, 42
Bandpass Filtering, 121
Bandpass Filters, 114
Block Code, 17, 171, 172, 173, 174
Boundary effects, 161
Broadband Level Meter, 56, 75
C
Calibrating Based on Microphone Sensitivity, 78
Calibrating with a Sound Level Calibrator, 77
Calibration, 77
Capture All, 32
Capture Measurement, 31
Capturing and Managing Stored Measurement Data, 31
Causes of Poor Coherence, 94
 Environmental noise, 95
 Problems with the measurement system, 95
 Reverberation, 95
Clarity Ratios (C35, C50, C80...), 126
Clock, 30, 56
Cloning Software, 172
Coherence, 93, 100
 Weighting, 17
Coherence Display, 94
Coherence Weighting, 17
Color Schemes, 30
Comma-Separated Values (CSV), 176
Common User Interface Elements, 24
Comparing Phase Traces, 90

Configurations
 Managing, 30
Configuring Input and Output Channels, 52
Configuring Live Measurements, 44
Configuring Smart for Real-Time Measurements, 44
Copy to ASCII, 36
Creating a new Live Measurement, 46
Creating and Managing Live Measurements, 45
Cursor Frequency Readout, 29
Cursor Readout, 27, 55, 111
Cursor Time Readout, 30, 114
D
Dante, 165
Data Display Controls, 112
Data Protection, 97
Deactivating an Installation, 171
Decibels (dB), 12
Delay Compensation, 84, 149
Delay Finder, 84
Delay Tracking, 85
Delete Many, 34
Delete Weighting Curve, 38
Deleting Stored Traces, 33
DFT, 5
Direct IR Measurement Using an Impulsive Stimulus, 138
Direct Sound, 107
Direct vs Indirect IR measurement, 137
Directional Loudspeakers and Reverberation Time, 143
Directivity Factor (Q), 156
Directivity index (DI), 157
Discrete Reflections, 107, 121
Distortion and Overload, 69
Dual Channel /Transfer Function Measurements, 80

- Dual Channel IR measurement
 - Using Period-Matched Signals, 139
- Dual Channel IR measurement
 - Using Random Stimulus Signals, 142
- Dual Channel IR Measurement
 - Using Random Stimulus Signals, 142
- Dual Channel Measurement and Display Configuration, 81
- Dual Channel Measurements, 80
- E**
- Early Decay, 107
- Early Decay Time (EDT), 109
- Early-to-late energy ratios, 109
- Early-to-Late Energy Ratios C50, C80 et al, 126
- Echoes, 159
- EDT, 109, 123
- educing Noise in IR
 - Measurements Made Using Random Stimulus Signals, 142
- Energy Time Curve (ETC), 119
- ETC, 119
- Evaluation Ranges (EDT, T20, T30), 123
- Examining Interaction Patterns with Spectrograph, 74
- Export Stored Data Trace as Weighting Curve, 38
- F**
- Fast Fourier Transform, 5
- Feedback Frequency
 - Identification, 73
- FFT, 5
 - Frequency Resolution, 6, 7
 - Time Constant, 7
 - Time Resolution, 6
- FFT Overlap*, 130
- FFT Size, 82
- FFT size and Averaging Controls, 113
- File-based Signals, 43
- Filter Settings, 116
- Finding Your Way Around in Smaart, 24
- Fourier Transforms, 5
- Frequency Display Settings, 116
- Frequency Domain Analysis, 128
- Frequency Resolution, 6
- Full Scale, 2
- Fundamental Concepts and Terminology, 4
- Further Reading, 151
- G**
- General Options, 29
- General Options Pertaining to IR Measurement, 114
- Global and Local Measurement Parameters, 48
- Global Settings
 - Spectrum Measurements, 48
 - Transfer Function, 49
- Glossary, 17
- Graph Area Allocation, 24
- Graph Controls, 26
- Graph Settings, 98
- Graph Type Selection, 25
- Grayscale (Spectrograph), 66
- Group Delay, 92
- H**
- Half Space, 161
- Hard Disk Cloning, 172
- Hide All Stored Measurements, 32
- Histogram Display, 127
- Histogram Settings, 116
- How to use this Guide, 1
- I**
- IFT, 5
- Import ASCII, 36
- Import Weighting Curve, 38
- Impulse Response
 - Analyzing, 117
 - Noise Floor, 107
 - Signal Presence Detection, 98
 - Uses, 108
- Impulse Response
 - Measurement Basics, 105
- Impulse Response Mode User Interface, 110
- Impulse Response Options, 115
- Indirect (Dual Channel) IR Measurement, 138
- Info Button, 35
- Installing Smaart v7, 167
- Introduction, 1
- IR Measurement
 - Averaging and Overlap, 148
 - FFT Size (Time window), 148
 - Measurement Duration, 147
 - Pushing the Button and Making the Measurement, 149
 - Recommended Default Settings for Dual-channel IR measurements, 150
 - Saving Your Work, 149
- IR Measurement
 - Additional Options, 114
 - Direct Measurement Using an Impulsive Stimulus, 138
 - Dual Channel Measurement Using Period-Matched Signals, 139
 - Dual Channel Measurement Using Random Stimulus Signals, 142
 - Excitation Level, 147
 - General Options Pertaining to, 114
 - Indirect (Dual Channel), 138
 - Input Levels, 147
 - Input source, 146
 - Logarithmic Sweeps, 141
 - Minimum Distance from Sound Sources, 145
 - Selecting Excitation Sources and Positions, 143

- Selecting Measurement Parameters, 146
- Selection of Measurement Positions, 145
- IR Mode
 - All Bands Table, 127
 - Bandpass Filtering, 121
 - Bandpass Filters, 114
 - FFT size and Averaging Controls, 113
 - Filter Settings, 116
 - Frequency Display Settings, 116
 - Frequency Domain Analysis, 128
 - Histogram Display, 127
 - Input Source Selector, 113
 - Level Markers (Ld, Le, Lr1, Lr2, Ln), 123
 - Logarithmic Time Domain Display, 118
 - Main Graph Area, 111
 - Measurement Controls, 113
 - Navigation pane, 110
 - Spectrograph, 129
 - Spectrograph Settings, 116
 - Time Domain Analysis, 118
- K**
- K versus k, 2
- L**
- Ld (Level Direct), 123
- LEQ, 75
- License Code, 168, 174
- Licensing and Installation, 166
- Line Arrays, 158
- Line Thickness, 29
- Linear and Logarithmic Frequency Scales, 10
- Linear and Logarithmic Scaling, 9
- Liner Amplitude, 11
- Live Averages, 47
- Live Impulse Response, 100
- Live IR, 96
- Live Measurement Controls, 58
- Ln (Level of Noise), 124
- Load Folder, 34
- Loading and Licensing the Software, 3
- Loading Stored Data from Files, 34
- Locked Cursors, 28
- Logarithmic Time Domain Display, 118
- Lr1 (Level Reverberant 1), 124
- Lr2 (Level Reverberant 2), 124
- M**
- Mac OS X, 167
- Machine ID, 3, 17, 21, 170, 171, 172, 174, 175
- Magnitude Averaging Type, 83
- Magnitude Response, 86
- Magnitude Thresholding, 97
- Main Display Controls, 57
- Main Graph Area, 56, 111
- Managing Audio Input and Output Devices, 50
- Managing Configurations, 30
- Material Absorption Coefficients, 153
- Measurement and Analysis Modes, 24
- Measurement Configuration Transfer Function, 82
- Measurement Group, 48
- Measurement Group Table, 48
- Measurement Groups, 45
- Measurement Parameters Global and Local, 48
- Measurement Rig Set-Up, 163
- Measuring an Acoustical Impulse Response, 137
 - What are we measuring, and why?, 137
- Measuring and Acoustical Impulse Response Direct vs Indirect IR measurement, 137
- Measuring Delays, 84
- Microsoft Windows, 167
- Migration Utilities, 171, 172
- Modulation Transfer Function (MTF), 133
- Mouse Cursor Tracks Data, 28
- Moving Smaart to a New Computer, 171
- MTW, 82
- Multi-Channel I-O, 164
- my.rationalacoustics.com, 174
- my.RationalAcoustics.com, 166
- N**
- Navigation pane, 110
- Network Audio, 164
- Noise Floor, 107
- O**
- Octave and Fractional-Octave Banding, 13
- On-Graph Menus, 26
- Operating System Upgrades, 173
- Options
 - Audio Device, 51
 - Impulse Response, 115
 - Transfer Function, 98
- Organizing your Data with Folders, 32
- Overlap (FFT)*, 130
- Overload Protection, 97
- P**
- Peak Hold, 62
- Phase, 99
 - Comparing Traces, 90
 - Unwrapping, 91
- Phase as Group Delay, 92
- Phase Response, 87
- Pink Noise, 41
- Pink Sweep, 42
- Pink Sweeps, 141
- Plot Calibrated Levels, 63
- Polar vs. Complex Averaging, 16
- Power vs Decibel Averaging, 16
- Propagation Delay, 107

pseudorandom noise, 42
Pushing the Button and Making
the Measurement, 149

Q

Quarter Space, 161
Quick Compare, 38

R

Reactivating after restoring
from a backup or after
migrating system files, 173
Real-Time Mode Main Window
Layout, 55
Real-Time Mode User Interface,
55
Real-Time Spectrograph, 65
Recapture, 32
Recommended Computer
Hardware, 3
Recommended Default Settings
for Dual-channel IR
measurements, 150
Reference Traces, 31
Capturing and Managing, 31
Reflections, 107, 121
Reformatting your System, 171
Registering a License, 167
Registering your installation, 3
Reporting Results for
Reverberation Time, 125
Reverberant
Build-up, 107
Decay, 107
Reverberation Time, 122
Frequency Ranges, 125
Level Markers (Ld, Le, Lr1,
Lr2, Ln), 123
Reporting Results, 125
Reverse Time Integration, 122
Room Volume, 153
RT60, 122
RTA
Plot Calibrated Levels, 63
RTA Display Settings, 68
RTA Graph Types, 62

RTA Measurements, 60
Run/Stop, 59

S

Saving Stored Data Traces to
Files, 33
Schroeder Cut-off Frequency,
154
Schroeder Integration, 122
Setting an Equalizer for a
Loudspeaker, 101
Setup Diagrams, 163
Show/Hide Live Measurement,
59
Showing and Hiding Stored
Measurements, 32
Signal Generator, 41, 57, 112
Signal Level / Sound Level
Meter, 56
Signal Presence Detection for IR
Measurements, 98
Sine and Dual Sine Waves, 43
Single and Dual-Channel
Measurement Techniques, 8
Single Channel Measurements,
60
Smoothing, 83
Smoothing (Transfer Function),
14
Software Installation
on Mac OS X, 167
on Windows, 167
Sound Level Calibration, 77
Sound Level Measurement: SPL
and LEQ, 75
Sound Level Meter, 112
Sound Level Options, 76
Spatial Averaging, 16
Spectrograph, 64, 129, 130
Buffer Size and Slice Height,
66
Dynamic Range, 66
Examining Interaction
Patterns, 74
Real-Time, 65

Spectrograph Analysis of an
Acoustical Impulse Response,
131
Spectrograph Basics, 64
Spectrograph Dynamic Range,
131
Spectrograph Settings, 69, 116
Spectrograph Time and
Frequency Resolution, 130
Spectrum Measurement Display
Configuration, 60
Spectrum Options, 67
General Settings, 67
Graph Settings, 67
RTA Display Settings, 68
Spectrograph Settings, 69
Speech Intelligibility Metrics,
132
Speech Intelligibility Modeling,
109
Speech Weighted noise, 42
Speed of Sound, 30, 115
SPL, 54, 75
SPL and LEQ Settings, 76
SPL/LEQ Logging, 77
Standards, 151
STI and STIPA, 132
Qualitative Thresholds, 135
System Migration Utilities, 171,
172
System Setup, 163

T

T20 and T30, 123
T60, 122
Tab-Delimited Text, 176
Text File Formats, 176
Thresholds, 26
Ticket File, 175
Time and Frequency Domain
Analysis, 4
Time Constant, 6, 7, 19, 23, 48,
49, 75, 76, 96, 100, 113, 129,
130, 131, 139, 142, 148
Time Domain Analysis, 118

Time Domain Display Settings,
115

Time Machine, 171, 172

Time Resolution, 6

Trace Information (Info), 35

Transfer Function

Graph Settings, 98

Transfer Function Control Strip,
81

Transfer Function Measurement
Configuration, 82

Transfer Function

Measurements, 80

Transfer Function Options, 98

Transition Distance, 159

U

Unwrapping the Phase Display,
91

Upgrading your Operating
System, 171

Upgrading your Operating
System or Major System
Components, 173

USB/FireWire, 164

Uses for impulse response
measurement data, 108

W

Weighting, 84

Weighting Curve, 37

Delete, 38

Export Stored Data Trace as,
38

Import, 38

What are we measuring, and
why?, 137

What is an impulse response?,
105

What is Smart?, 1

Z

Zooming, 26