

Lime 9.16 Documentation Addendum



Annex to the Lime™ Manual
for version 9.16.99.
(work in progress)
2020-11-05



Table of Contents

1. Introduction.....	10
1.1. Summary	10
1.2. System Requirements	11
1.2.1. Windows	11
1.2.2. Apple Macintosh	11
1.2.2.1. Warning for MAC Users.....	11
1.3. Web Site	11
2. Registration	11
3. Annotations	12
3.1. Annotation Selection - Right-Click.....	12
3.2. Unicode Support	12
3.2.1. Windows Code Pages	12
3.2.2. Unicode Fonts.....	12
3.3. Annotation Anchoring	13
3.3.1. Horizontal Anchoring	13
3.3.2. Vertical Anchoring	14
3.3.2.1. Chord Ornament-Fixed Anchoring.....	14
3.3.2.2. System Anchoring.....	15
3.3.2.3. Staff Spacing Scalable Style.....	15
3.3.2.4. Line/Curve Vertical Growing.....	16
3.3.3. Text Justification	16
3.3.4. Text Annotation Placement Dialog.....	16
3.3.5. Line Placement Dialog.....	17
3.3.6. Resizable (Extended) Lines and Curves	18
3.3.6.1. Resizable Line and Curve Selection.....	19
3.3.6.2. Clearing Extendability.....	19
3.3.6.3. Anchoring Extendable Lines and Curves.....	19
3.3.7. Annotation Placement Experimental Dialog Short-Cuts	19
3.4. Pasting Text into Annotations	20
3.5. Constrained Drawing and Moving	20
3.5.1. Shift when Creating a Line or Curve	20
3.5.2. Shift when Modifying the Size of a Line or Curve (or Shape)	21
3.5.3. Shift when Dragging an Annotation	21
3.5.4. Alt when Dragging an Annotation (high resolution, slow dragging)	21
3.6. Fonts	21
3.7. Backslash Codes in Annotations	22
3.7.1. New Backslash Codes	22
3.7.2. Backslash Editing Option	23
3.7.3. Backslash Editing Mode	23
3.8. Stack Vertically and Fingering.....	23
3.9. Fingering and String	24
3.10. Auto-generated Text Annotations	26
3.11. Multi-Line Text	27
3.12. Undo and Redo	27
3.13. Extending Underlines and Hyphens in Lyrics.....	27
3.13.1. Terminating Underline Extension	27
3.14. Staff Size Adjusting	28
3.15. Annotation Assistants	28
3.15.1. Chord Symbol and Fred Grid Assistants	29





3.15.2. Old Chord Symbol Sizing	29
3.16. Annotation->Text (Category) Menus.....	29
3.16.1. Text and Marl+reset	29
3.16.2. Annotation Classifications	30
3.16.2.1. Figured Bass	30
3.16.2.2. Poet and Dedication Annotation Classification	30
3.16.3. Playback Interpretation.....	30
3.17. Lines and Curves	30
3.17.1. Rectangle Drawing	30
3.17.2. Line Clipping.....	30
3.17.3. Extended Curves	30
3.17.4. Line and Curve Flipping.....	31
3.17.5. Curve Inversion.....	31
3.17.6. Curve Line Width.....	31
3.17.7. Line Staff Size Adjusted	31
3.18. Partially Visible Annotations.....	31
3.19. Duplicated Annotations.....	31
3.19.1. Duplicate in Every Part Context.....	32
3.19.1.1. Invisible Duplicated Annotations.....	32
3.19.2. Deleting Duplicated Annotations.....	32
3.19.3. Modifying Duplicated Text Annotations	32
3.19.4. Redundant Duplication.....	33
3.20. Group Selection	33
3.21. Allow Any Graphic Alt-Right-Click	34
4. Hearing.....	35
4.1. Short Cuts	35
4.2. Hear Dialog	35
4.2.1. Parts to Hear	36
4.2.2. Begin at the Beginning	36
4.2.3. Metronome	36
4.2.3.1. Metronome and Time Signature	37
4.2.3.2. Metronome Channel.....	37
4.2.4. Tracking.....	37
4.2.5. Skip Repeats.....	37
4.2.6. Do Not Reset Midi	37
4.2.7. Ignore Sustain Ped	37
4.2.8. Use Piano if Voice has no Instrument	38
4.2.9. Hear to End of Piece.....	38
4.2.10. Voices in Context	38
4.3. Default MIDI Instruments (Programs).....	38
4.3.1. Piano Sounds	38
4.4. Resetting MIDI	39
4.4.1. Detecting New MIDI Devices.....	39
4.4.2. No Sound.....	39
4.5. Playable Pitch Range.....	39
4.6. Microtones, Pitch Bend and Portamentos	39
4.7. Modify Playback and Re-compute Playback.....	40
5. Short Cuts	41
5.1. Menu Short-Cuts	41
5.2. Changed Default Short Cuts	41
5.3. User Defined Short Cuts	41
5.4. TAB and Number Pad Arrow Key Short Cuts	42





6. Pre (and post) 9.15 Scores	43
6.1. Upgrading Issues	43
6.1.1. Annotations on Invisible Rests	43
6.1.2. Invisible Rests become Visible	43
6.1.3. Stem Directions	43
6.1.4. Annotation Horizontal Placement	43
6.1.5. Page Width	44
6.1.6. Tuplet Plotting	44
6.1.7. Mark and Accent Placement	44
6.1.8. Old Scores between Macs and Windows	44
6.1.9. Fonts on Macs	44
6.2. 9.17 Format	45
7. Import and Export	46
7.1. MusicXML Importing	46
7.1.1. Unicode Support	46
7.1.2. Creator Funnies	46
7.1.3. Notation Ornaments	47
7.1.4. Fingering and String Specification	47
7.1.5. Dynamics and Micro-Tones	47
7.1.6. Tuplets	47
7.1.7. Slurring	47
7.1.8. Tempo Marking	47
7.1.9. Importing Options	48
7.2. MusicXML Exporting	50
7.3. MusicXML File Name Extensions	50
7.4. NIFF Import	51
7.4.1. SharpEye	51
7.5. NIFF Export	51
7.6. Scanned Music	51
8. Duration Error Fixing	52
8.1. Fixer Contexts	52
8.1.1. Voices in a Fixer Context	53
8.1.2. Ticker-Tape	53
8.1.3. Oddities	53
8.1.4. Saving	54
8.1.5. Redundancy	54
8.2. Duration Editing	54
8.3. Finding Duration Errors	55
9. Colours	56
9.1. Music Window Appearance	56
9.2. Colour Palettes	56
9.2.1. Customisation	57
9.2.2. Inverted Colours	57
9.3. Colouring Note Heads and Keys by Note Name	57
9.3.1. Default Note Name Colour Mapping	58
9.4. Colouring Issues	58
10. System and Staff Distance Layout Options	59
10.1. Absolute Staff/System Separation Parameters	59
10.2. Additional Before Staff/System Separation Parameters	59
10.3. Staff Drag	60





11. Parameters	61
11.1. Default Values	61
11.2. Normal Values	61
11.3. Character Values	61
11.4. Note Head Widths	61
11.5. Line Widths	61
11.6. Parameter Notes	62
11.6.1. Y offset from top staff line to number of bars	62
12. Zoom Layout	63
12.1. Default Zoom	63
13. Bar Lines	64
13.1. Hidden Bar Lines	64
13.2. Hearing Options	65
13.3. Concatenated Bar Lines	65
14. Slurring	66
14.1. Slur Association	66
14.2. Adding Slurs and Ties	66
14.3. Grace Slur	66
14.4. Slur End Placement on Stem or Note Head	67
14.5. Ties as Slurs	67
14.6. Conversion of Old or New Files	67
15. Hidden Objects	68
15.1. Hidden Annotations	68
15.2. Hidden Symbols	68
15.3. Hidden Rests	68
15.4. Hidden Clefs	68
15.5. Hidden Key Signatures	68
15.5.1. Default Accidentals	69
15.5.2. 8 ^{va} and 8 ^{vb}	69
15.5.3. Showing Invisible Key Signatures	69
16. Preferences, Context Options and Piece Options	70
16.1. New Preferences	70
16.1.1. Make ~Backups when Saving	70
16.1.2. No Warnings when Opening Older or Newer File Formats	70
16.1.3. Emboldening Preference	70
16.1.4. Metronome is default when Hearing	70
16.1.5. Highlight Notes with Unplayable Pitch	71
16.1.6. Old Default Accidentals	71
16.1.7. Music Window Flashing Rate (of insertion point)	71
16.2. New Context Options	72
16.2.1. Automated Footer	72
16.3. New Piece Options	72
16.3.1. Uniform Marks as in 9.05 and earlier	72
17. Polymetric and Polytemporal Parts	73
17.1. Polymetric Parts Option	74
17.2. Note Editing	74
17.3. Beaming	74
17.4. Polytempo Note Duration Multiplier (very advanced feature)	75





17.4.1. Re-computing Playback	76
17.4.2. Linear Spacing	76
17.4.3. Upgrading Old Scores.....	76
17.5. Systems.....	76
17.5.1. Bulk Breaks with the Page->Systems... option	76
17.5.2. Manual Insertion of System and Page Breaks	77
17.5.3. Measure Insertion and Deletion	77
17.5.4. Further Study	77
18. Parts and Voices.....	78
18.1. Parts and Voices... Dialog	78
18.1.1. Arrangement Propagation	78
18.1.2. Apply	78
18.1.3. Previous and Next.....	78
18.1.4. Don't Print	78
18.1.5. Move Staff.....	78
18.1.6. All Staves	79
18.1.7. No Ledger Lines	79
18.1.8. Changes to Number of Staves.....	79
18.1.9. Part or Voice Change	79
18.2. Idle Staff Hiding.....	79
19. Quarter Tones and Special Accidentals	81
19.1. Quarter Tone Special Accidentals Dialog	81
19.2. Note Entry	83
19.3. Hearing Special Accidentals	83
19.4. Special Accidentals Parameters.....	83
20. Ticker-Tape (panorama) and Autocue Contexts.....	84
20.1. Setting Up a Ticker-Tape or Autocue Context	84
20.2. Navigation.....	85
20.3. Nominal Page Width of a Ticker-Tape Context.....	85
20.4. Nominal System Separators in a Ticker-Tape Context	85
20.5. Page Size & Zooming	85
20.6. Editing	86
20.7. Annotation Anchoring	86
20.8. Printing	86
21. Miscellaneous Items.....	87
21.1. About Lime.....	87
21.2. Edit->Clear	87
21.3. Dialogs	88
21.3.1. Entering Numbers in Dialogs	88
21.3.2. Tri-State Check-boxes.....	88
21.4. Note Entry	88
21.4.1. Grace Notes	88
21.4.1.1. Nachschlag Grace Notes.....	88
21.4.2. Default Accidentals.....	89
21.4.3. Transposing Instruments.....	89
21.4.4. Replacing Tuplets by Other Tuplets	90
21.5. Key Signature Dialog	90
21.5.1. Short Cut to Set Sharps or Flats	90
21.5.2. Default Accidentals.....	90
21.5.3. Transposing Instrument	90





21.5.4. Octave Shifted Play-Back.....	91
21.5.5. Deleting Key Signatures.....	91
21.5.6. Showing Naturals Before Bar Line.....	91
21.6. Clefs 8 ^{va} and 8 ^{vb}	91
21.7. Transposing Parts.....	91
21.8. Showing Triplet Numbers.....	92
21.9. Parenthesized Accidentals.....	92
21.10. Piano Window.....	92
21.10.1. Piano Window Mode.....	93
21.10.2. Piano Window Music Mode Buttons and Status Indicators.....	93
21.10.3. NUM-Lock Indicator.....	94
21.10.4. Piano Window Zoom menu.....	94
21.11. Rulers.....	94
21.12. Low Resolution Rendering.....	94
21.12.1. Printing.....	94
21.12.2. Copy Rectangle.....	95
21.13. Time Signatures.....	95
21.13.1. Advanced Time Signatures.....	95
21.13.2. Placement.....	95
21.13.3. Polytemporal.....	95
21.14. Measure Insertion and Deletion.....	96
21.15. Symbol Placement at System Breaks.....	96
21.15.1. Bulk Changes - Systems Option.....	96
21.15.2. Manually Splitting Systems.....	96
21.15.3. Clef, Key Signature and Time Signature Dialogs.....	96
21.16. Next, Previous Buttons and Apply Buttons in Symbol Dialogs.....	97
21.16.1. Apply.....	97
21.16.2. Previous and Next.....	97
21.17. Saving.....	97
21.17.1. Locking (Publishing) Scores.....	98
21.17.2. Undo After Save.....	98
21.17.3. File Size.....	98
21.18. Go To Bar and Go To Page.....	98
21.18.1. Going to Beginning of Current Page.....	99
21.18.2. Going to First or Last Page.....	99
21.19. Go Back to last selection.....	99
21.20. Merging Staves.....	100
21.21. Rhythm Errors and Voice Length Consistency Checks.....	100
21.21.1. Next Rhythm Error.....	100
21.22. Don't Share Note Head.....	100
21.23. Switching Contexts.....	100
21.24. Select All.....	101
21.25. Beaming over Rests.....	101
21.26. Multi-Length Staff Chords.....	101
21.26.1. Accumulated Arpeggios.....	102
21.27. Braces.....	103
21.28. Midi File Export.....	104
21.29. Midi File Import.....	104
21.30. Lime Console.....	104
21.31. Navigation when Inspecting Internal Fields.....	104





21.32.	Bar Lines Through First Chord (advanced feature)	105
21.32.1.	Experimental	106
21.33.	System Breaks	106
21.33.1.	System Breaks from List	106
21.33.2.	Ignoring Timing Option	106
21.33.3.	Re-Pagination	107
21.33.4.	Staff Drag	107
21.34.	Pasting Music	107
21.35.	New Note Heads and Accents	107
21.36.	Symbols at End of Staff	107
21.37.	Fixing Missing Reminders	107
21.38.	Note Stem Flags	108
21.39.	Guitar Tablature	109
21.40.	Batch File Processing	110
21.40.1.	Destination Specification	110
21.41.	Shift Key Menu Alternatives	111
22.	Max OSX Considerations	112
22.1.	OSX Version Support	112
22.1.1.	Warning	112
22.1.2.	Mojave, 10.14	112
22.1.3.	Yosemite, 10.10	112
22.2.	Printing	112
22.2.1.	Print Quality	112
22.2.2.	Post-Script Printing Deprecated	112
22.2.3.	Print Centering	113
22.2.4.	Copy Rectangle deprecated	113
22.3.	QuickTime and Hearing Pitch Bend and Microtones	114
22.4.	Long File Names	114
22.5.	Fonts Dialog	114
22.6.	Asian and Middle Eastern Annotations	114
22.7.	Oriental Users	115
22.8.	Zoom Limits	115
22.9.	Music Font Annotations	115
22.10.	Connecting MIDI Devices	115
23.	Windows Considerations	116
23.1.	Windows Version Support	116
23.2.	Refreshing Windows	116
23.3.	Multi-Monitor Environment	116
23.3.1.	Mouse Click on Lime Window Ignored	116
23.3.2.	Lime Does Not Appear at Start up	116
23.4.	Gestures on Windows	117
23.5.	Windows Font Smoothing	117
23.6.	Right to Left and East Asian Languages	118
23.7.	PDF Printing	118
23.8.	Ctlr+AltGr	118
23.9.	Bad Midi Devices Crashing PC	119
23.10.	File Name Problems on Windows	119
23.10.1.	Exotic File Names (fixed in 9.16)	119
23.10.2.	Leading Spaces in File Names	119





23.11. VMware Virtual Machine	119
24. Lime Lighter	120
24.1. Manual Scrolling	120
24.1.1. TAB and Return Keys	120
24.1.2. Hearing	121
24.2. Automatic Scrolling	121
24.3. Full Screen Mode	121
24.4. Mark-Up	122
24.4.1. Printing Marked Up Pieces	122
24.5. Ticker-Tape and Autocue Contexts	122
24.5.1. Mark-Ups	123
24.5.2. Switching Between Normal and Ticker-tape or Autocue	123
24.6. Lime Lighter Specific Short Cuts	123
24.7. Lime Lighter Clicking & Gestures in Music Window	124
24.7.1. Configuration of 1 & 2 finger Tap Gestures	124
24.7.2. Press and Tap Gestures	125
24.8. Lime Lighter Preferences	125
24.8.1. Pedal Mapping	127
24.8.1.1. Old Infinity Pedal	128
24.8.1.2. Alternative Pedals	128
24.8.2. Lime Lighter Parameter Defaults	129
24.8.3. Lime Lighter Arrow Keys	129
24.9. Lime Lighter on Macs	129
25. Lime Aloud	130
25.1. Lime Aloud - Special Speaking Options	130
25.2. Toggling Lime Speaking	130
25.3. Hearing Status	130
25.4. Progress Messages	130
25.5. Commas and Periods	131
25.6. Navigation Short-Cuts	131
25.7. JAWS Issues	131
25.7.1. Dialogs	131
25.8. Alternatives to JAWS	131
26. Lime-GoodFeel Braille	132
26.1. Lime-GoodFeel Preferences	132
26.2. Braille Window	132
26.2.1. Braille Font	133
26.3. Braille Timeout	133
26.4. Archaic GoodFeel 3.2	133





1. Introduction

In the interests of releasing improved versions of Lime, upgrading the manual has had to be postponed. This document comprises draft text for inclusion in the Lime manual. As such, it is an essential component of the Lime 9.15 and 9.16 documentation. It describes enhancements and changes up to and including version 9.16.

In general, with the exception of annotation anchoring, the use of Lime 9.15 and 9.16 is upwards compatible with earlier versions.

The last 9.16 release is 9.16.99. This includes most features referred to in this document as 9.17.

1.1. Summary

Main changes from 9.0x (amongst over 700 enhancements and fixes) are:

- New file format to support the new annotation anchoring capabilities and larger zoom sizes. The new format can be read and converted back to the previous format by 9.14.9 but not anything earlier. This format is twice the (9.0x) size on disk.
- Truly high quality printing on Intel Macs (without postscript).
- New, flexible annotation anchoring capabilities, including bar lines and the right of the page. Also supports right justified text.
- Right-Click replaces Alt-click (which is deprecated) for selecting and manipulating annotations. Alt-right-click is still needed for the 'Extended Alt-Click' option. N.b. on Macs, Ctrl-click is tantamount to a right-click.
- Batch update facility, enabling multiple files to be updated to the new format or to be changed to templates.
- Major improvements in MusicXML import. However export is still very basic. Improving MusicXML importing is a never-ending project.
- Command-period (Ctrl-period on windows) replaces command/ctrl-H for invoking the hear option. However, you can still set command/ctrl-H as a short cut.
- Ctrl-Alt-letter is now available on Windows to provide 26 additional user settable short cuts. On Macs, Command-Alt-letter replaces Ctrl-letter.
- On Macs, pressing Ctrl while selecting a menu item can be used instead of Shift (this replaces the use of Alt, which is no longer available).
- New backslash expressions for editing annotations; includes \>> and \<< which replace \1 and \2 which are now used for MARL numeric symbols.
- White on Black & Black on Yellowy display options plus other colour improvements.
- Performance improvements, such as formatting a page of music, or pasting music.
- Improvements to Lime Lighter, including full screen display, ticker-tape & autocue contexts, automatic scrolling, go to bar touch screen navigation, and a Low Vision parameter template set for display.
- 9.15.8 and later has the ability to edit Unicode annotations saved by post 9.16 Lime. 9.16 creates Unicode annotation if needed. 9.17 will have full Unicode support.

The final 9.16 re-release (9.16.99.) recognizes all initial 9.17 features. It is, therefore, recommended that all users install the re-release, while waiting for the 9.17 official release.





1.2. System Requirements

1.2.1. Windows

Lime 9.16 is available for Windows XP and later. It has been tested up to Windows 10.
Whilst it still works, support for Windows 2000 is deprecated.

1.2.2. Apple Macintosh

Lime 9.16 is available for Apple Mac OSX 10.4 and later. It has been tested up to 10.14, Mojave.

Lime still works on old Power PC Macs, however support for Power PCs and for OSX 10.5 (Lion) and 10.4 (Tiger) is deprecated.

1.2.2.1. Warning for MAC Users

Apple have announced that the last system to support 32 bit applications, such as Lime, is Mojave (MacOS 10.14). If you wish to continue using Lime, do not upgrade to Catalina (MacOS 10.15) or later.

This is unhelpful as, unfortunately, Apple have made it particularly difficult to upgrade old programs to 64 bits. It will therefore require a lot of resources, which we do not have, to upgrade Lime.

1.3. Web Site

Lime's main web site is still at www.cerlsoundgroup.org. Visit the site to check for and download new versions.

Note that www. web sites can usually be invoked with or without the leading www. However occasionally, domain name server (DNS) problems can stop one of them working. So, for any web site, including Lime's, if it doesn't work with www, try it without.

2. Registration

Basic Lime is now 'donation-ware'. Anyone who previously purchased it, is deemed to have 'donated'. Donations, which go towards the development of facilities for blind and partially sighted musicians, may be made at www.dancingdots.com/lime/LimeDonate.html.

Those who have donated will receive a registration key. Once this has been entered, the initial splash page that is displayed when Lime starts up will no longer appear. All Lime facilities are available, whether or not you have donated. The only exception is that the largest zoom is disabled if you haven't registered.

Visit Dancing Dots www.dancingdots.com for more information on Lime based products for blind and partially sighted musicians. N.B. Dancing Dots requires a separate registration process for customers who order one or more of the special accessibility features marketed as Lime-Lighter, Lime-Aloud, and GOODFEEL[®]. Do not try to enter a licence key for one of these products into Lime directly.





3. Annotations

Annotations, whether text, lines or curves, are often referred to as "*annos*" within Lime.

3.1. Annotation Selection - Right-Click

Annotations are selected or moved by using the right mouse button - *Right-Click* (on Macs, *Control-Click* can also be used). This replaces Lime's original *Alt-Click*, which is deprecated. If any '*allow graphic alt-click*' option is turned on, *Alt-Right-Click* must be used to select and convert a standard graphics element.

Though it is deprecated, alt-click can still be used to select an annotation; but the alt-key must be released when dragging an annotation, if you do not want high resolution.

3.2. Unicode Support

In 9.16, when editing an annotation, if the text cannot be rendered without Unicode, it will be converted to Unicode. However, by default, if it can be rendered without Unicode it will be. This also applies to annotations in MusicXML or NIFF imports.

In Lime 9.17, Unicode annotations will be fully supported (entailing a minor change to the file format). Until then, 9.15.8 and all later versions are capable of handling files containing Unicode annotations (they can recognize, display and edit them).

As part of the transition to full Unicode support, in 9.16.5 and later, there is a piece option to specify that Unicode should be used wherever possible for annotations in the piece. When setting Unicode support, there is an option to convert the piece to use Unicode (whenever possible) or otherwise.

It is strongly recommended that, unless you are a user of GoodFeel 3.5 or earlier, Unicode is used, particularly if the piece contains foreign language annotations (any language requiring diacritics or non-roman characters). Use of Unicode is also recommended if there is any chance of a Lime score being read on both Windows and Macintosh. This is because Unicode is the only reliable, compatible encoding.

3.2.1. Windows Code Pages

Prior to Unicode, Windows used code pages to distinguish different scripts (or character sets). Scores created on Windows with annotations using non-ANSI code pages will not render properly on Macs unless they are first converted to Unicode on Windows.

Note that on Windows for compatibility, until Unicode is fully supported, if you use a non-English language for entering text (e.g. Greek, Russian, etc), if it can be rendered using a valid code page, it will be. If it cannot be rendered using a code page, or if multiple languages are used in the same annotation, Unicode will be used.

3.2.2. Unicode Fonts

On both Macs and Windows, if an annotation's font does not contain the necessary glyphs for a character, the system uses a font fall-back or chaining algorithm to identify a suitable font to use instead. Unfortunately there is little or no consistency (particularly on Windows). Therefore, for Middle Eastern and Asian languages, it is recommended that an appropriate font is used, that contains the necessary glyphs (e.g. *SimSun* for Chinese). A useful font for multi-lingual text is *Arial Unicode MS*, which is available on most modern systems.





3.3. Annotation Anchoring

One of the main enhancements in Lime 9.15 and later is flexible annotation anchoring. Apart from anchoring on the note-head, previous versions limited the horizontal anchoring to the left of the page (called 'horizontal lock'), and the vertical anchoring to the staff middle ('called 'vertical lock') and the page bottom (but only combined with 'horizontal lock'). Lime now supports independent anchoring horizontally and vertically, enabling annotations to be anchored not only to the note-head but also to the page (left, right, centre, top, bottom, and middle), to the staff (left, right, and the vertical middle), and to the previous and next bar lines.

All annotations are placed at fixed distances from the horizontal vertical anchor points. In different contexts these anchor points may be at a different place on the page but the annotations will be at their designated distances from them. Furthermore, if anything is changed, such as page size, system formatting, or even the music itself, annotations will remain at their designated position from their anchor points.

3.3.1. Horizontal Anchoring

<i>Anchor</i>	<i>Relativity</i>	<i>Description</i>
Note	Head	The anchor point is the left edge of the note-head. This is the default.
	Accidentals on Chord	The anchor point is just to the left of any accidentals on the staff chord in which the annotation's note participates. Typically the annotation will be right justified (and have the 'leave-space' style). This is useful for placing things such as an arpeggio.
Page	Left	The left edge of the page; (this used to be called 'horizontal lock').
	Center	The page centre. Title, etc, are typically anchored to the page centre.
	Right	The right edge of the page. Such annotations will, typically, have right justification
Staff	Left	The left edge of the current staff
	Right	The right edge of the current staff (usually right justified)
Bar Line	Left	The bar-line to the left of the annotation's note. If there is no bar-line, the end of the preamble (clef, key and time signatures, etc) at the left of the staff is used.
	Right	The bar-line to the right of the annotation's note. If there is no bar-line, the staff-right is used.





3.3.2. Vertical Anchoring

<i>Anchor</i>	<i>Relativity</i>	<i>Description</i>
Note	Head	The anchor point is the middle of the note-head. This used to be the default, however nowadays, chord-ornament anchoring may be more useful.
Page	Top	The top edge of the page. Typically titles, credits, etc will be anchored to the page-top.
	Middle	The page middle. This is for completeness. It will probably not be much use unless pictures are supported.
	Bottom	The bottom edge of the page; (this used to be called 'footer')
Staff	Middle	The middle line of the current staff; (this used to be called 'vertical lock'). Typically, lyrics, directives and labels will be staff-relative.
System <i>after 9.16.8</i>	First staff	The middle line of the first (top) staff in the system
	Last staff	The middle line of the last (bottom) staff in the system
Chord Ornament	Above	Above any automatically placed notation symbols, above the staff
	Fixed.	Fixed as a chord ornament (cannot be moved vertically). See below for further information.
	Under (below)	Below any automatically placed notation symbols, under the staff

3.3.2.1. Chord Ornament-Fixed Anchoring

Chord Ornament-Fixed anchoring can be convenient as a mechanism for entering ornament annotations for multiple contexts, when a voice appears with different stem directions and, possibly, differ whether ornaments (such as Fermata) are placed above or below the staff.

The *Chord Ornament-Fixed* anchoring was designed to facilitate MusicXML import and the automatic placement of notation symbols above or below the staff, depending on voice layout, in the absence of any indication in the MusicXML (as is common).

Care must be taken when using *Chord Ornament-fixed* anchoring because there are some limitations with the associated automatic placement which include:

- Done in order found.
- Any alt-click generated annotations are always analysed after the others.
- While an annotation with a *Chord Ornament-fixed* anchor is being edited, any other annotations, with *Chord Ornament* anchoring on the same note, may move. When editing is complete, they will be restored to their original position.





3.3.2.2. System Anchoring

Anchoring to the top or bottom is useful for annotations which are relevant to all parts in a system. It is available in 9.16.9 and later.

System anchoring is available in 9.16.9 and later. It is recognized by Lime 9.16.8 but not by any version prior to 9.16.8.

3.3.2.3. Staff Spacing Scalable Style

As of 9.16.9, annotations can have a style flag that scales the vertical position to the actual spacing between staves. The scalability flag is, currently, set using the *Annotation Placement* dialog, using the *scalable* check-box

Staff Spacing Scalability is available in 9.16.9 and later. It is recognized by Lime 9.16.8 but not by any version prior to 9.16.8.

If an annotation has the “*Staff Spacing Scalable*” style, the vertical position of the annotation will be scaled according to the vertical separation between its staff and the next one below on the page.

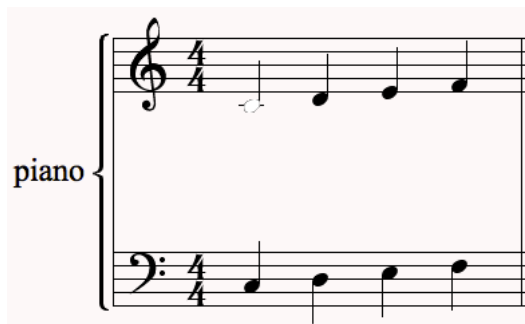
The *Staff Spacing Scalable* style cannot be applied to annotations that are anchored to the page or are a fixed chord ornament. Any attempt to do so will be ignored.

The adjustment is based on the ratio between a nominal separation (80 points) and the actual separation from the previous staff on the page when the page is formatted, which is derived from:

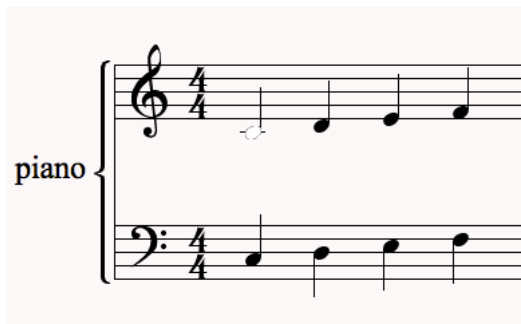
- The default separation specified in the context layout...
- Adjustments made by staff/system separation parameters.
- Vertical compression of the page if it is too dense to fit.

For the last staff in any system (including the last on the page), the adjustment is based on the separation to the first staff of the next system (if there is one). For all other staves, the adjustment is based on the separation to the next staff below in the system. Thus, if you want to place an adjustable annotation between two staves on a system, it should be on the upper staff.

A common use of staff spacing adjustment is to place labels on grand staves. If such a label is placed on the upper staff, positioned mid way between the two staves, and the *Staff Spacing Adjust Style* is set, it will always remain roughly at that midway position, whatever the actual separation.



Unscaled, 80pt separation



Scaled, 60pt separation





In the *Annotation Placement* dialog, the vertical offset is always the unscaled value.

Vertical Positioning Anchor

☐ Offset

☒ scalable

In this example of a label placement, the nominal separation is 80pts, so the mid-way distance is 40pts. In general, the middle of a text annotation is about a quarter of it's font size above its placement. Thus the label is placed at 3pts (=12/4) below the mid-way position.

3.3.2.4. Line/Curve Vertical Growing

If a curve or line or line based shape has the *Staff Spacing Growable* style then the vertical height is also scaled with respect to the staff separation. The style is, currently, set using the *Line Annotation Placement* dialog, using the *growable* check-box:

Line/Curve
Vertical Positioning Anchor

☐ Offset

☐ scalable ☒ growable

Line/Curve Vertical Growing is available in 9.16.9 and later. It is recognized by Lime 9.16.8 but not by any version prior to 9.16.8.

Growable lines' clip points (and curves' control points) remain at the same distance from their respective ends, scaled by the staff size. This means that, by carefully setting the position and clip points, a *growable* line can be used for a line that is drawn between staves (like bar lines), whatever the staff separation or staff size.

3.3.3. Text Justification

Text annotation may be specified as being left-justified, right-justified, or 'centre-justified'.

There is also a special justification, known as '*Centered*', which not only centres the annotation at its horizontal anchor¹ (see section on annotation anchoring), but also prevents it being moved horizontally. Its main use is for lyrics.

Because justification and anchoring are closely related, text justification is specified in a unified dialog² along with the anchoring, which is invoked using the '*Annotation Placement...*' menu item from the *Annotation* menu.

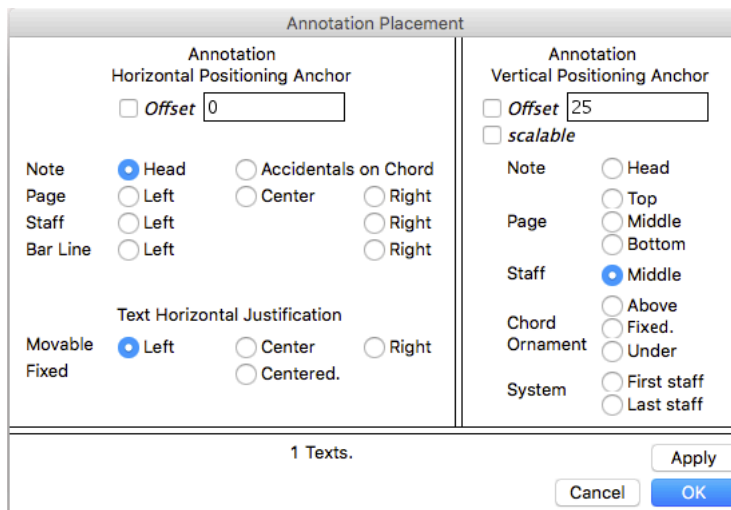
3.3.4. Text Annotation Placement Dialog

The anchoring (and text justification) is specified using the '*Annotation Placement...*' menu item from the *Annotation* menu. This is a unified dialog, which enables horizontal and vertical anchoring to be specified as well as the justification of the selected text annotations. The predefined shortcut for this dialog is in Windows Control-, (comma) and on Macintosh Command-,.

¹ actually 'Centered' annotations are 3pts to the right of the left of the note head, which is effectively the centre of the note-head.

² for compatibility, the old *style->centered* menu option now automatically invokes this dialog, with justification preset to centered.



The dialog box is titled "Annotation Placement". It is divided into two main sections: "Horizontal Positioning Anchor" and "Vertical Positioning Anchor".

Horizontal Positioning Anchor:

- ☐ Offset: 0
- Note:** ☒ Head, ☐ Accidentals on Chord
- Page:** ☐ Left, ☐ Center, ☐ Right
- Staff:** ☐ Left, ☐ Right
- Bar Line:** ☐ Left, ☐ Right
- Text Horizontal Justification:**
 - Movable:** ☒ Left, ☐ Center, ☐ Right
 - Fixed:** ☐ Centered.

Vertical Positioning Anchor:

- ☐ Offset: 25
- ☐ scalable
- Note:** ☐ Head
- Page:** ☐ Top, ☐ Middle, ☐ Bottom
- Staff:** ☒ Middle, ☐ Above, ☐ Fixed., ☐ Under
- Chord:** ☐ First staff, ☐ Last staff
- Ornament:** ☐ Last staff
- System:** ☐ Last staff

At the bottom, it says "1 Texts." and has buttons for "Apply", "Cancel", and "OK".

Text Annotation Placement Dialog

When the dialog is first invoked, it shows the anchoring justification of the selected annotation(s). If more than one annotation is selected, radio buttons are highlighted if any annotation in the selection has that characteristic. If one is clicked, then all annotations will get that characteristic. If any of the selected annotations is text then the dialog is oriented towards text placement and the justification section is included

Click the relevant radio button to select the required anchoring or text justification. Once you click a radio button, all other buttons in the group are cleared.

The *Offsets* show the placement of the annotation(s) in points from the anchor (horizontal and vertical). If there is more than one annotation and the offsets are not the same, the offset field will be blank initially. The offset value can only be changed if the associated check box is checked.

If an offset value is changed, it will be set for all selected annotations. This can be useful as an alternative way of aligning annotations.

3.3.5. Line Placement Dialog

If none of the selected annotations are Text, then the dialog is oriented towards Lines and Curves and, if any are resizable, includes a section for the horizontal anchoring of the (right) end of the Line(s)/Curve(s).





Line/Curve Placement Dialog

The example in the figure above shows the anchoring (and positioning) of an extendable line that always extends from the beginning of the bar to the end of the bar in which its corresponding note is.

If more than one line/curve is selected, changes to the End Horizontal Positioning only affect the resizable ones.

3.3.6. Resizable (Extended) Lines and Curves

When the “Extend Line or Curve to Here” facility is used, the end of a Line or Curve is initially anchored to the note-head of the specified note:

- Normally to the left (offset 0) or, if the Shift key is used, to the right;
- For right brackets, to the right or, if the Shift key is used, to the left.

This will usually cause the end of the line/curve to move. The position can, of course, be changed by dragging the end of the line/curve; the anchor can be changed using the *Annotation->Annotation Positioning* dialog.

If a line/curve style is set to Extended using the *Annotation->Line and Curve Style* menu, the end of the line/curve will initially have the same anchoring as the beginning of the line and the end of the line/curve will not move.

Like text annotations, the anchoring for a line or curve annotation is specified using the ‘*Annotation Placement...*’ dialog. If all the selected annotations are lines or curves, and one or more are extendable, this dialog also allows specification of the anchoring of the end of the end of the line/curve.

When a line/curve is extended over a system boundary, the first 2/3 of the line/curve appear both before the end of the system and the last 2/3 of it appear after the beginning of the system. If the curve goes back on itself³, the duplication can become undesirable; in which case, it may be necessary to use two separate curves.

³ A ‘complex curve’ is one whose 1st Bezier control point is significantly to the right of its 2nd one.





3.3.6.1. Resizable Line and Curve Selection

A resizable line/curve may be selected by right-clicking near one of its real end points. This applies even if it extends over a system or page break. However right-clicking on the right of the first part of the line before the break or on the left of the last part of the line after the break has no effect; similarly, if the line extends over 2 or more systems, right clicking on one of the middle sections has no effect.

If a line/curve annotation extends over a system break, the selected part is fully highlighted and may be manipulated; the other end is partially highlighted to indicate where it is, but cannot be manipulated.

If a line/curve annotation extending over a system break is selected by right-clicking the end that was clicked will be the selected part and the associated note will become the insertion point and flash. This means that, by right clicking the end, one can see which is the associated end note.

3.3.6.2. Clearing Extendability

A resizable/extendable line/curve may be converted back to normal using the *Annotation->Line and Curve Style* menu. If it extends over a system or page break, the beginning part will be resized to a default length, which, in most cases, will need adjusting.

If, for some reason, the terminating note of an extendable line/curve is deleted, the line/curve will be converted automatically normal. This can happen if the note is part of a sequence of notes replaced by another, longer note or rest, or if it is deleted when measures are deleted.

3.3.6.3. Anchoring Extendable Lines and Curves

A resizable/extendable line/curve cannot be vertically anchored to the page. This is because, if it extends over a system break, the extension will still be at the same vertical position and the effect can be strange.

3.3.7. Annotation Placement Experimental Dialog Short-Cuts

The Annotation placement dialog has an experimental short-cut mechanism, whereby two letters may be used to 'click' a radio-button or check box, providing one is not editing one of the offset values. The first letter selects a block (an asterisk will be displayed before the block name after the first letter is pressed); the second letter selects the radio-button or check box within that block. If the second letter is the same as the first, the short cut is cancelled.

Note that, when the dialog is first invoked, the initial focus is on this invisible short-cut 'edit' field.

h	Horizontal Positioning Anchor
hn	Note head (also h.)
ha	note Accidentals on chord
hp	Page left
hr	page Right
hs	Staff left (also h[)
hg	staff riGht (also h])
hb	Bar left (also h\)
hi	bar riGht (also h/)
ho	horizontal Offset





- v Vertical Positioning Anchor
 - vn Note head (also v.)
 - vt page Top
 - vm page Middle
 - vb page Bottom
 - vc sCalable
 - vs Staff middle
 - va chord ornament Above
 - vx chord ornament fiXed
 - vu chord ornament Under
 - vf First staff on system
 - vl Last staff on system
 - vo vertical Offset
- j Justification (Texts only)
 - jl Left justified
 - jr Right justified
 - jc Center justified
 - jx Centered and fixed.
- e End Horizontal Positioning Anchor (Lines/Curves only)
 - en Note head (also e.)
 - ea note Accidentals on chord
 - ep Page left
 - er page Right
 - es Staff left (also e[)
 - eg staff riGht (also e])
 - eb Bar left (also e\)
 - ei bar rIght (also e/)
 - eo horizontal Offset

3.4. Pasting Text into Annotations

When text is pasted into an annotation (while it is being edited), the font and style of the pasted text is also included, unless it is explicitly pasted unformatted. Pasting unformatted is achieved by using the shift key (e.g. Ctrl-shift-V or, on Macs, Cmd-shift-V).

3.5. Constrained Drawing and Moving

When moving (dragging) any annotation or when drawing or modifying a line or curve, the Shift key can be used to constrain the direction. This can be particularly useful for drawing horizontal or vertical lines or drawing squares.

3.5.1. Shift when Creating a Line or Curve

A line or curve is created by selecting Line or Curve mode, then with Command/Ctrl key down, press the (left) mouse key and dragging at the required position.

When the Shift key is active, the direction will be constrained to horizontal, vertical, or diagonal, depending on the initial direction of movement, with the following exceptions:





- Rectangles, of any type, are always constrained to squares (i.e. the movement is constrained to a diagonal).
- Hairpins are always constrained to a fixed height and to the horizontal. This makes it relatively straightforward to draw hairpins with uniform heights.

In addition, if the Shift key is active at the beginning, and the line/curve is note head anchored, and the starting horizontal position is near the selected note head, it will be snapped to the left of the note head. To do this, first hold down the Command/Ctrl and Shift keys together, then press the mouse key and start drawing; the shift key can be released after starting to draw, to remove any directional constraints.

3.5.2. Shift when Modifying the Size of a Line or Curve (or Shape)

If the shift key is active when dragging one of the end points of a Line or Curve, the initial aspect ratio will be maintained. This means that the angle will stay the same or, if it is a rectangle, the proportions will be the same.

Hairpins are an exception; if the Shift key is active, when modifying their size, the hairpin height will remain constant and only the width may be changed.

3.5.3. Shift when Dragging an Annotation

If the Shift key is down while dragging an annotation (right-click-drag), the direction of movement will be constrained to either **horizontal** or **vertical**, depending on the initial direction. For this to be effective, the Shift key must be pressed **AFTER** the (right) mouse button, otherwise it will be treated as a group selection.

3.5.4. Alt when Dragging an Annotation (high resolution, slow dragging)

Since 9.16, the Alt (option) key, not the Shift key, is used for high-resolution adjustments. It is, therefore, important that the ALT key is released when dragging an annotation (or drawing a line/curve), if high resolution is not needed.

3.6. Fonts

In order to ensure a degree of compatibility between systems, since 9.15, Lime files contain accurate information about the fonts used in the piece. This allows the score to be read on any Windows or Mac OSX system. Prior to that, the font mapping, if any, was erratic and there was no mapping before version 9.0. This means that, unless well known semi-standard fonts were used, there is nothing in old files to indicate what actual fonts were used.

When reading old files, if there is no mapping information, Lime attempts to identify the fonts that may have been used. If identification is impossible, then *Times New Roman* will be used to display it.

When reading a Lime score, Lime will check that the fonts, which are identified in the score, are installed on the user's system. If they are not installed, then an appropriate default will be used for display purposes. In most cases, the default is *Times New Roman*, but in a few cases *Arial Narrow* or *Arial* is used (e.g. for the old, narrow *Avant Guard*).

If an annotation using an uninstalled font is edited, then for that annotation, the uninstalled font is automatically converted to the font used for display purposes.

The *Edit->Clear->Substitute All Uninstalled* Fonts menu item will convert all annotations using uninstalled fonts to the font used as a substitute.





3.7. Backslash Codes in Annotations

Backslash expressions are used when editing annotations to create special (music) characters and to specify formatting style. The font for such expressions should always be a simple, Western alphanumeric font, otherwise if the annotation is not flagged as Unicode (supported in 9.16) it may get converted when editing.

A backslash expression comprises a `\` and 1 or more case significant characters, optionally followed by a delimiting semi-colon; E.g. `\q`; means a quarter note in the Marl font, `\+`; means a sharp (*for a chord symbol*), and `\++`; means a double sharp (`\+`;+ is sharp followed by +).

When preparing an annotation for editing, in order to avoid any confusion between a backslash expression and the character after it, unless the expression is followed by another expression or a space, Lime will delimit the expression with a semi-colon.

3.7.1. New Backslash Codes

The following are new or modified backslash codes [not described in the manual]:

<code>\</code>	Backslash followed by a space is a fixed width, “non-breaking” space character. It can be useful, for example, when stacking ornaments, including a delayed turn, represented by a non-breaking space followed by the turn character in the Marl font (<code>~</code>). In simple fingering, it keeps the space with the next character.
<code>\\</code>	Backslash followed by another backslash is a single backslash. This allows a backslash to appear in an annotation.
<code>\N</code>	Backslash followed by N (capital-N) is a new-line. Text following it will be on a new line (see section 3.8).
<code>\R</code>	Backslash followed by R (capital-R) is a carriage return. Text following it will start at the beginning of the same line, overlaying the existing text.
<code>\;</code>	Reverts font size and style to that in force at the beginning.
<code>\>></code>	Decrease font size by 4 points; replaces <code>\1</code> . (<code>\1</code> is now number 1 in a music font).
<code>\<<</code>	Increase font size by 4 points; replaces <code>\2</code> . (<code>\2</code> is now number 2 in a music font).
<code>\></code>	Decrease font size by 1 point.
<code>\<</code>	Increase font size by 1 point.
<code>\<></code>	Sets the base font size (for subsequent <code>\></code> , <code>\<</code> , etc) to that of the last character.
<code>\,</code>	A thin-width (non-breaking) space.
<code>\.</code>	A dot (as in a dotted note).
<code>\..</code>	A double dot.
<code>\ </code>	A vertical line
<code>\`</code>	A slash (e.g. as used in figured bass)
<code>\l</code>	Toggle <u>underline</u> (like <code>\b</code> and <code>\i</code> toggle bold and <i>italic</i>).
<code>\v</code>	Toggle subscript (like <code>\u</code> toggles ^{superscript}).
<code>\FONT</code>	Reverts to the base font (e.g. after <code>\MARL</code>).
<code>\niente</code>	A niente circle (e.g. for diminuendo to nothing).
<code>\arp</code>	The arpeggiation character.
<code>\darrow</code>	Down arrow suitable for creating a Down arpeggio (e.g. <code>\arp\R\darrow</code>);





`\uarrow` Up arrow suitable for creating an Up Arpeggio (e.g. `\arp\R\uarrow;`).

`\0 .. \9` `\` followed by a digit means the equivalent digit in a music font, as used, for example, for numbering full bar rests.

Note that the sizing codes (`\<<`, `\>>`, `\<`, `\>` and `\<>`) are all with reference to a standard, 5pt staff, whether or not the annotation is actually adjusted for staff size.

3.7.2. Backslash Editing Option

If backslash codes are used when originally creating or modifying an annotation, the 'backslash editing' style will be set for the annotation. This means that if the annotation is edited, the original backslash expressions will be used. This can be changed using the 'backslash editing' option from the annotation style menu. Chord symbols and other annotations using superscripting should always be edited using the backslash option.

There is also a preference option to always use backslash editing. By default this is on if Lime's low-vision, speaking option is set, and off otherwise.

Care must be taken if setting backslash-editing for an annotation that was originally created without backslashes. Whilst the result, if left unchanged, should remain the same, the backslash expressions that Lime creates, may not match the original intention and can be a little confusing. This can also happen if character parameters used by the annotation have since been changed.

If an annotation was originally created when the preferred music font was MARL/TUFA and subsequently edited in backslash mode when the font is SONATA (or vice versa), symbols in the old font will be converted to the prevailing preferred music font.

3.7.3. Backslash Editing Mode

As well as the codes for music symbols, backslash codes include options to toggle face modes, **bold**, *italic* and underline and to change the font size. When an existing annotation is edited in backslash mode, the face and the size of text is first set to that of the first character. Text face toggling and font size backslash code start with reference to that.

Due to the potential conflict between the style (face, size, etc) set using backslash expressions and that set by the user using the annotation style menu, if a backslash expression is found that changes the style (currently face or size only), then any equivalent change in the actual text-edit record is ignored. So if `\b`; `\i`; or `\l`; is ever encountered, the text face of any subsequent text face of any subsequent characters is ignored. Similarly if `\>>`; `\>`; `\<<`; or `\<`; is ever encountered, the font size of any subsequent characters is ignored. This behaviour can be changed in the preferences.

3.8. Stack Vertically and Fingering

If the *Stack Vertically* (*Annotation->Style-> Stack Vertically*) style is set, providing there are no explicit new-lines, the text is split into new lines at each character (except dashes, slashes, parentheses, braces and brackets, carriage return or non-breaking space).





In addition, Style->*Fingering* can be used to set or clear vertical stacking and, optionally, to set or clear fingering classification⁴ and to set the various defaults (size, style, etc) for fingering annotations. Using Style->*Fingering* to set fingering is tantamount to setting the classification with Annotation->Text with Marl+defaults->*Fingering*.

Typically the special characters (dashes, slashes, parentheses, etc) will be used as follows:

- dashes Used to specify a finger change (substitution).
- / slashes Used to specify an alternative
- () parentheses Used to specify the string that should be used.
Braces {} and brackets [] are also treated specially

3.9. Fingering and String

With Lime 9.17 comes the ability to set fingering and string information specific to each note. A fingering annotation can be specified as automatically generated from the fingering data in the staff or stem chord that the note is part of.

The fingering data allows for alternate and/or substitute fingers and associated strings. A finger can be one of:

- a Number up to 5⁵;
- the conventional guitar *PIMA* letters ('p', 'i', 'm', 'a', and 'c' or 'e');
- t for thumb;

A string identifier can be one of:

- a Number from 1 to 8;
- a Note Letter (one of cdefgab)

Up to 4 fingers and/or strings can be specified. This allows allow for fingering such as 1/2-2/3 (*1 or 2 substituted by 2 or 3*), with a string specification for each. The most common use, however, will be just 1 finger. The scope of note related fingering is deliberately restricted so that it is well structured for MusicXML import and for for Lime Aloud and/or brailleing purposes. If anything more sophisticated is needed, ordinary annotations, classified as fingering, can always be used.

A note's fingering data are context independent. They are printed by special fingering annotations, whose text is automatically generated. As, like all annotations, these can be context specific, it is possible to show fingering in some context and not others; for example fingering can be shown in a part's context but not in the score.

The fingering and/or string for a note is set using the fingering helper which is in the Note Menu (*Note->Fingering...*). Normally one will want not only to set the fingering but also to display it in an annotation; however the fingering helper does provide an option not to make or modify annotations. These annotations are auto-generated (see section 3.10), so the contexts will change if the associated data changes. If you require the fingering to be shown in just one context, then use that context for setting the fingering and making the annotations, not the main Score context.

⁴ Note that programs such as GoodFeel require the classification to be Fingering in order to handle fingering annotations properly.

⁵ Lime assumes the user has no more than 5 fingers per hand ☺






Basis use of the Fingering Helper

The basic use of the fingering helper shows a basic finger (1) for the selected note.

If the note is actually part of a staff chord, whose other notes also specify fingering, then the resultant annotation can show the fingering for all notes in the conventional stacked manner. When making auto-generated annotations, one has the option to specify whether the annotation is generated from just the note, or from all or part of the chord (see section 0).

It should be noted that the resultant annotations are backwards compatible in so far as that 9.16 and earlier will treat them as ordinary fingering annotations.

The fingering helper dialog only allows valid fingers and strings to be used, as defined above. Any attempt to enter more than one character or an improper one, will be rejected (with a beep).

The < and > buttons will apply the specified fingering (if there is any change and the *Shift* key is not down) and move to the previous (<) or next (>) note in the staff chord. This is designed to allow fingering to be specified on successive notes without having to re-invoke the dialog.

If the *More...* checkbox is checked, alternate or substitute fingering can be specified. An example is shown in the following figure. Also shown is an extreme example of the possible use of alternate and substitute fingering and associated strings. The associated text that will be generated in an annotation is shown above the *Reset* button.

Alternate Fingering specification

Extremely Complex Fingerings and Strings





3.10. Auto-generated Text Annotations

With Lime 9.17 comes the ability to define text annotations to be automatically generated from data about the note with which they are associated. Currently these data are:

Fingering and String. Note associated *Fingering* and/or *String* are described in section 3.9.

Note name. The *Note Name* is simply the basic basic name of the associated note and is one of CDEFGAB. Normally one will not want the note name except, possibly, for preparing scores for education purposes.

The text for an auto-generated annotation has one line for each of the associated notes. Normally, particularly if there is more than one note, the annotation will be specified as Stacked, as is usual with fingering. Rests are ignored.

An annotation can be specified as Auto-Generated using the *Annotation->Auto Generated...* menu option.

Creating an Auto-Generated Annotation

If the Automatically Generated checkbox is checked the annotation will be auto-generated, otherwise any auto-generation will be cleared.

An auto-generated annotation has an association, which can be one of:

Staff Chord. All non-rests in the Note's staff chord.

Stem Chord. All non-rests in the same stem direction in the Note's staff chord.

Voice Chord. All notes in the Note's voice chord

Note Only. Just the Note. This can be useful if you want fingering or string to be displayed next to note-heads, rather than above or below the staff.

An auto generated annotation can be specified to be coloured like the associated Note if automatic note colouring is turned on.

Auto-generated annotations are backwards compatible in so far as that 9.16 and earlier will treat them as ordinary fingering annotations.





3.11. Multi-Line Text

Support for multi-line text (created by using `\W` backslash code) is limited; it is primarily aimed at simplifying MusicXML import. The following should be noted:

- Style options such as extending underscore or hyphens are not available with multi-line text.
- If the *Stack Vertically* style is set, then new-lines go above the previous line if the annotation is above the note or staff; furthermore, if there are new-lines (`\W`) or carriage-returns (`\R`), splitting is only at those characters. Leading and trailing new lines are not plotted if the *Stack Vertically* style is set. A leading new-line can, therefore, be used to force new line based stacking.
- Like other annotations, the selection points for multi-line text are top-left, top-right, bottom-left and bottom-right. Right-clicking anywhere else in the text will do nothing (unless it is near another annotation).

To edit a multi-line text, you should place the cursor on the *1st line*, then edit the whole text annotation as a single line. If the annotation is wide, when editing one or both ends could extend off the sides of the music window and will scroll horizontally automatically if necessary. Remember that the arrow keys can be used to navigate forwards, backwards, to the beginning and to the end.

3.12. Undo and Redo

As of 9.15.5 there is basic support for undo and redo while editing text annotations⁶. When editing text the Undo and Redo menu options (normal short cuts are Ctrl/Cmd-Z and Ctrl/Cmd-shift-Z), will undo or redo typing, pasting, font changes, etc within the text.

3.13. Extending Underlines and Hyphens in Lyrics

When entering lyrics, it is normal to extend underlines and centre hyphens at the end of a chord's phrase to the next chord. The annotation styles, *Center Hyphens* and *Extend Underlines* control this. An annotation comprising a single underscore extends any existing extension further to the next one. The underline and hyphens must be the last character in a style annotation. If they are followed by a space, then the effect will not happen.

3.13.1. Terminating Underline Extension

When an extended phrase ends on a note, the extended line is normally aligned flush with the right of the last note-head; the exact position being governed by the parameter:

LYRICS AND TEXT

Right end of underline extends this far under ending note (7).

⁶ Prior to 9.15.5 Undo, while editing text would simply exit text edit mode, discarding all changes.





Currently (9.16) there is a problem if the extension goes over a system break; it just doesn't appear after the break unless there is a leading underscore annotation on the first chord after the break. However an annotation comprising just an underscore and a non-breaking space⁷ has the effect of terminating an extending underline, whether or not it extends over a system break. It will be placed correctly if it is *Center-Justified* (not *Centered*) with the horizontal offset set to the negative of the right end parameter (-7). As a convenience, if `_\\` is entered while typing lyrics (*Centered* annotations with *Extend Underlines*), the justification and offset will be set automatically.



Extending Underline over System Break

3.14. Staff Size Adjusting

When a text annotation's style is set to '*Staff size Adjust*', if the voice is on a different sized staff in another context (or if the staff size is simply changed), the font sizes of the annotation will be scaled proportionally to the different staff size.

N.b. if you do need to edit such an annotation, it is recommended that, if possible, you edit it in the original context with the original staff size, particularly if backslash expressions are used. This is because occasionally, due to the need to round font sizes to discrete values, resizing back to the original can be 1pt out. For example, 14pt text on a 5pt staff, reduces to 8.4pt on a 3pt staff, which rounds down to 8 pt. When resized up again to 5pt, it increases to 13.3333pt, which rounds to 13pt, which is 1pt lower than the original!

3.15. Annotation Assistants

If an annotation assistant dialog is invoked while a text annotation is being edited, the whole annotation will be replaced.


⁷ A non-breaking space is entered using *back-slash space* (see back-slash expressions, 3.7) or just a single *back-slash* at the very end.





3.15.1. Chord Symbol and Fred Grid Assistants

The chord symbol and fret grid assistant dialogs include an option to set the base font size. This is the font size appropriate to the current staff size. It defaults to the prevailing default for the category, if one has been set.

Both dialogs also include an *Apply* button to apply and see the result, without leaving the dialog. They also include 2 navigation buttons to go to the next or previous chord. This allows a sequence of chords or fret grids to be inserted without leaving the dialog. Enter the required symbol, click *Apply*, navigate to the next chord , enter the next symbol, click *Apply*, etc.

3.15.2. Old Chord Symbol Sizing

The '*old sizing*' option for chord symbols, used additional superscripting for accidentals used in the alteration component, that were the default in 9.15 and earlier, and also the slightly larger, pre-9.16 'quality' sizes. If you are entering new chords in a piece with existing chords, you might want to set the '*old sizing*' option in order to make the new chords look the same as the old ones. The differences are shown in the example below.

New sizing: Cmaj(#5^b9) Old sizing: Cmaj(#5^b9)

3.16. Annotation->Text (Category) Menus

If a text category is selected from the *Annotation->Text* menu, annotation characteristics are set to those appropriate to the selected category and text entry is initiated (in Text mode). If the font or other characteristic is changed, whilst that category is selected, any new annotation will be created with the same characteristics.

If a new category is chosen, the prevailing defaults for that category are always set. Similarly, whenever any single annotation is selected, its characteristics are set for the associated text category.

With the exception of *Fred-Grid*, which requires the Tufa font (indicated by the semi-colon before it), the default font for all annotation categories is Times, when using the normal Text Category menu. This allows use of the back-slash notation without needing to change fonts.

For some categories, the Marl font can be used by default if the menu entry is chosen while pressing the SHIFT-key. This applies to the *Articulation*, *Technical/Bowing*, *Dynamic*, *Ornament*, *Pedal*, *Repeat* (ending) and *Pause* and categories.

3.16.1. Text and Marl+reset

The alternative Text menu behaves the same as the normal Text menu with the following differences:

- If there is no user defined model for the category, the attributes will all be set to the default.
- The Marl font will be used by default for editing the *Articulation*, *Technical/Bowing*, *Dynamic*, *Ornament*, *Repeat*, *Pedal* and *Pause* categories. These are labelled with an asterisk (* star) in the menu. This can be overridden using the SHIFT-key.



3.16.2. Annotation Classifications

3.16.2.1. Figured Bass

There is now a 'figured bass' annotation classification, which defaults to below the staff. A single figured bass annotation can be created using the \N new-line facility.

3.16.2.2. Poet and Dedication Annotation Classification

The Poet/lyricist and Dedication annotation classifications are similar to the composer classification. The default placement for poet/lyricist is on the left of the page, and for dedication it is in the centre.

3.16.3. Playback Interpretation

Some of the classes of annotation can effect play-back, and will default to being interpreted. These include:

- Tempo Marking
- Dynamic
- Midi

Note that, if an annotation's style is set to "*Do Not Interpret*", then there will be no effect on play-back, whatever it might contain. Thus an annotation, such as:

Program 72 (Clarinet)

will only set the midi instrument if interpreted. If you expect an annotation to affect play-back, but it doesn't, check that it does not carry the "*Do Not Interpret*" style.

3.17. Lines and Curves

3.17.1. Rectangle Drawing

The line mode options "Rectangle" allow drawing of rectangles and generalised vertical or horizontal brackets (similar to hairpins, etc). Squares can be drawn by using the Shift key while drawing.

Like other 'meta-lines' (hairpins, etc), rectangles' selection points are the top-left and bottom-right. However, you can also select them by right- (or alt-) clicking on the bottom-left or top-right.

3.17.2. Line Clipping

All lines have two clip points associated with them. These clip points allow you to clip parts of the line; the clipped portions of the line will not be visible.

If the line is a bracket of any sort, an end with a bracket cannot be clipped vertically. If a line is a hairpin, vertical clipping is constrained to one of the actual lines that form it.

3.17.3. Extended Curves

Curves can be extended in the same way as Lines. If a curve extends over a system break, 2/3rd of it will be before the break and 2/3rd after. If the curve is not a simple slur-like curve, the result will need to be checked.



3.17.4. Line and Curve Flipping

Using one of the 'flip' items, the *Annotation->Line and Curve style* menu, lines and curves annotations may be flipped horizontally or vertically. This adjusts the positioning (and clipping) and/or the meta-style to change the appearance.

3.17.5. Curve Inversion

Curves can be inverted using the *Annotation->Curve* menu. Inversion simply flips the curve's Bezier control points. Invert Vertical will, for example, convert a curve up into a curve down, and vice versa (this is the primary purpose of the facility). For complex curves, you will have to see what the effects are.

3.17.6. Curve Line Width

As of Lime 9.16.6, the line width values set in the *Annotation->Line and Curve Style* menu can be also be used for curves. In addition, the option to specify that a curve is fattened in the middle, has been moved to the *Annotation->Curve* menu. Note that non-standard sized curves will be rendered as normal sized in earlier versions of Lime.

3.17.7. Line Staff Size Adjusted

If a Line or Curve annotation is *staff size adjusted*, its vertical location is adjusted according to the size of the staff with which it is associated. This is the same as staff size adjusted text annotations. If it is also *growable*, its height will be adjusted according to the staff size, rather than the staff separation.

If any annotation's vertical location is specified as *scalable* (i.e. with respect to staff separation) that takes priority over staff size position adjustment.

3.18. Partially Visible Annotations

The selection points of an annotation are normally at each end of it; when an annotation is selected, they are shown as little black rectangles. If one or both ends are outside of the music window, then the selection points will be at the edge of the music window and, when selected, will be just visible. Furthermore, if an annotation's placement is such that it would be completely outside the music Window, Lime will ensure small portion is visible

3.19. Duplicated Annotations

It is recommended that "*Duplicate at Every System*" and "*Duplicate on Every Page*" are only used on annotations that are specifically in only one context, unless all contexts will always retain their current system (and page) structures. If an annotation in, say, *All Contexts*, is duplicated *at Every System*, if a context changes its system structure, the annotation may be removed and/or appear in funny places in other contexts.





3.19.1. Duplicate in Every Part Context

If you want an annotation to appear in all contexts (parts), but none of the contexts have a common voice on which it can be inserted, then *Duplicate in Every Part Context* can be used (see the main Lime manual⁸). Common uses of *Duplicate in Every Part Context* include duplicating tempo markings and titles so they appear in all parts' contexts.

If *Clear in Every Part* is used, the result depends on what is selected:

- | | |
|--------------------|---|
| Notes | If one or more notes are selected, then <u>all</u> annotations attached to the selected notes, which have been duplicated in every part, are cleared in all other contexts. |
| Annotations | If one or more annotations are selected, then only annotations with the same classification and have been duplicated in every part, are cleared. |

In both cases, only the duplicates are cleared, the selected remain.

3.19.1.1. Invisible Duplicated Annotations

Annotations *duplicated in every part context* are actually duplicated into every voice. In each context, only such annotations on a voice on the first staff of the associated system are actually shown. However the others are still there. If the visible result of a duplication is deleted in a context, only that one annotation is deleted. If there are other voices on the system, the annotations will still be there, albeit invisible (but may be visible in other contexts)

Where there are many measures that are just rests, Lime will combine the measures, if there is no impediment. The presence of an annotation, whether visible or not, will prevent multi-measure rests being created. If there is a problem with combining multi-measure rests, *Symbol->Hidden Annotations...* can be used to temporarily show hidden and invisible annotations (including duplicates).

3.19.2. Deleting Duplicated Annotations

If an annotation, that has been duplicated, is deleted (cleared), then the user is asked whether the duplications should also be deleted. If the shift key is used when invoking the deletion (e.g. shift+backspace), then all duplicates are deleted with no warning.

In the case of *Duplicate in Every Part* annotations, duplicates are only deleted if the change is made in the score context. If a duplicate is deleted from any other context, none of the other duplicates are affected.

Clearing a "*Duplicate at Every System*" and "*Duplicate on Every Page*" annotation deletes all associated annotations, including the selected one(s), without any question.

3.19.3. Modifying Duplicated Text Annotations

If the contents of a duplicated text annotation is modified, the user is asked whether the modification should be propagated to all the duplicates. As with deletion⁹, in the case of *Duplicate in Every Part* annotations, propagation only happens if the change is made in the score context

⁸ In the old manual, *Duplicate in Every Part* used to be also called *Duplicate in Every Context*.

⁹ Prior to 9.16.9, deleting or modifying a duplicated annotation had no effect on any of the duplicates.





3.19.4. Redundant Duplication

If an annotation is duplicated when it has already been duplicated, the results are redundant. Such redundant annotations are silently cleared when a piece is read from disk. The "*Edit->Clear->Redundant Duplicate Annotations...*" can also be used.

3.20. Group Selection

When multiple annotations are selected using right (or alt) mouse drag, the annotations that are actually selected depend on the annotation mode. If the shift key is used, then all annotations in the selection rectangle are selected¹⁰. However, if the shift key is NOT down, then the annotations selected depend on the starting annotation mode, as displayed in the Piano Window:

- If the starting mode was Music mode or a group of annotations is already selected, then any annotation in the rectangle is selected.
- Otherwise, only those annotations that match the current mode will be selected. So, for example, if in text mode, only text annotations can be selected, if the shift key is not down.

When a group of annotations are selected, the mode icon highlighting in Piano Window reflects the types selected. If more than one type is selected, then more than one icon will be highlighted. If you want to extend a group selection and you wish to constrain to the type(s) selected, start the mouse drag with the shift key down, but then release it.

If you want to use right (or alt) mouse drag to group select just one type of annotation, exit any group selection then select the appropriate icon in the Piano Window before dragging the mouse.

¹⁰ This was the behavior prior to 9.16.9, even when the shift key was not used.



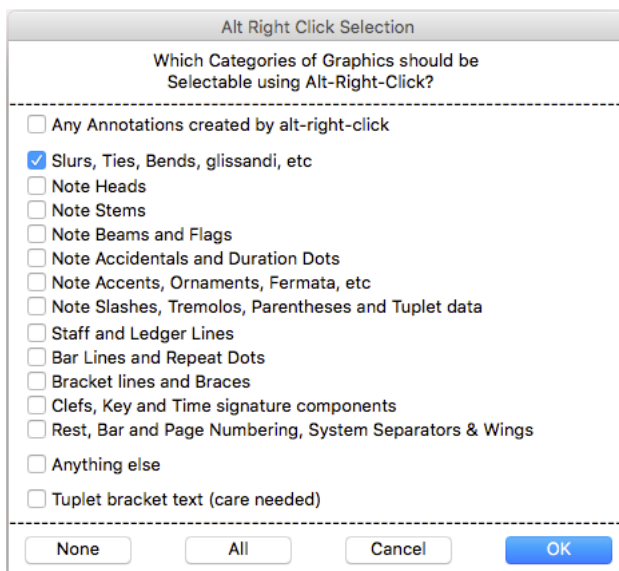


3.21. Allow Any Graphic Alt-Right-Click

The “*Allow Graphic Alt-Right-Click*” item in the edit menu allows one to **Alt-right-Click** on a Lime generated graphic or text and convert it to an annotation that can be moved or edited.

Note that **Alt** must be used for special graphic conversion; right click by itself will never do it.

The menu item invokes a dialog, which allows one to specify the categories of graphic that can be selected. One should select only the category that is actually needed. Even then care should be taken to select the graphic actually required.



Alt Right Click Selection dialog

If “*Allow Graphic Alt-Right-Click*” is invoked with Shift, it will toggle the last selection off and on.

By and large modifying Lime’s graphics should not be required. The one exception is to adjust slurs and ties, particularly if a slur involves another staff (a feature that Lime does not yet support).

Apart from the checkboxes themselves, the **None** button will clear all selections and the **All** button will set them (in the unlikely event that that is required). Two checkboxes are slightly special:

- If ‘*Any Annotations Created by Alt-Right-Click*’ is checked then one can use **Right-Click** to select any category of graphic that has already been converted to an annotation. If it is not checked, then only those in the checked categories may be selected
- If ‘*Tuplet bracket text*’ is checked, one can also **Alt-right-Click** the tuplet text of a tuplet bracket and make it a separate annotation. It should be noted that, if such an annotation is decomposed, there can be occasional strange side effect, so it should be used with caution.

Group selection is also possible with **Alt+right button**. If group selecting with **Alt-right-button**, normal annotations will be ignored (unless the shift key is down as well).

Alt-right-Clicked annotations behave just like any other annotations, except that they do not prevent rests being combined. Also, if a converted Line is extended and set resizable (recommended for things like converted slurs and ties), it becomes completely independent and the original is deleted.

If an **Alt-right-Clicked** annotation is deleted, then neither it nor the original graphic is visible. The original graphic can be restored using **Edit->Clear->Kill Annotation**. All **Alt-right-Clicked** annotations can be cleared from a note, by selecting the note and using **Edit->Clear->Restore-Alt->Annotations**.





4. Hearing

4.1. Short Cuts

The short cut to invoke the 'Hear' option has been changed to Control-period (command-period on Macs). On Windows, the old Control-H is also set by default to invoke the hear menu. On Macintosh, Command-H is NOT set by default, because it is the system standard for hiding the application; however it can be set if required. On both systems, *Control/Command-period* may always be used to invoke the Hear dialog or to pause hearing, whether or not another short cut has been defined.

If play-back is inactive, *Control/Command-period* invokes the Hear dialog. If used with the shift key, the dialog will default to beginning at the beginning.

If play-back is active, *Control/Command-period* by itself will suspend play-back; if it is pressed again, play-back will resume at the beginning of the bar in which it was suspended. If used with the alt-key (option), play-back is stopped completely. If used with the shift-key the Hear dialog is always invoked, stopping any play-back.

If play-back is active **with tracking** (only when tracking), *Control/Command-]* will play faster and *Control/Command-[* will play slower. Also *Control/Command-P*, if set to the default (Print), is deactivated while hearing because, if visually challenged, it is too easy to hit 'P' instead of '[' or ']'.

4.2. Hear Dialog

The Hear dialog that controls the hear option, serves two purposes:

1. To specify the parameters for hearing and to invoke the hearing.
2. To apply parameters for subsequent hearing, including automatic hearing (in Lime Lighter or when hearing to the end of piece, using ctrl/cmd+down-arrow).

Hear Option Dialog

If you press 'OK' in the Hear dialog, then play-back will start.





If you press '*Apply Done*', then the options chosen will just be saved as the default for next time or for any automatic playing. When a piece is saved, the latest set of hear options are saved, so that they are pre-set next time the piece is loaded. If you press '*Apply Done*' with the shift-key down, the options are remembered, but the dialog remains and you can then press '*OK*' to start play-back.

4.2.1. Parts to Hear

In the Hear option dialog, one can specify which parts (from those in the context) will be heard. These parts are saved with the score file. There are two sets of parts:

normal: the set used for normal hear options.

auto: the set used for hearing when Lime Lighter scrolling (manual and automatic) and when using ctrl/cmd+down-arrow. Lime Lighter users, who use automatic scrolling, should use this facility to set up what, if any, parts play when playing while scrolling (the '*auto*' set of parts) and how the metronome is used.

It is possible for a set to be completely empty. This can be useful (particularly for automatic Lime Lighter scrolling) when one just wants to see the scrolling music and, possibly, hear metronome clicks, but nothing else.

4.2.2. Begin at the Beginning

The 'Begin at the Beginning' check box specifies whether Lime should play from the beginning of the piece (or from the current bar). If the Hear menu is invoked with the SHIFT key down, this will be set by default.

4.2.3. Metronome

Lime supports the use of a metronome both as a lead-in when hearing and while actually playing. There is a check box specifying that Lime should include a metronome click while playing the piece. By default this is off, except for Lime Aloud users, but the default can be set using the general preferences.

There is also a check box specifying whether there should be Lead-in metronome clicks before the actual playing begins.

Lime works out from the first time signature how many beats to a bar there should be and an appropriate number of lead-in beats. In general it will beat on the denominator of the time signature. However, if a compound or additive time signature is explicitly specified, the beat will be on each component of the compound. For example, 6/8 will normally beat 6 quavers to a bar; however if it is specified as 3+3/8 (use compound radio box in the BAR dialog), it will only beat 2 dotted crotchets (quarter notes) to the bar. Complex additive rhythms, up to 4 components in the bar are supported.

If starting at the beginning of a measure and it's not a pick-up measure, the lead-in will be the normal number of beats in the first measure; there is an option to double this. If there is a pick-up measure (see section 13), or if starting in the middle of a measure, the lead-in will be the number of beats up to the first beat in which the note appears. If there is only one such beat, then it will click for a complete measure plus this beat.

The 'Lead out' checkbox specifies that, if sounding the metronome while playing and the end of piece is reached, the metronome will continue to play until the end of the prevailing rhythm, or one more measure. If partially checked, it will play just one additional beat.

If a metronome is chosen, there are options to specify what it sounds like (including option for different first beat).





4.2.3.1. *Metronome and Time Signature*

The metronome requires that there is a time signature specified on the reference voice. If there is no time signature, there can be no metronome, neither lead-in nor while playing (except on the very first beat). If you do not want a time signature, but do want a metronome, then use a hidden time signature with the required values.

An accurate metronome, of course, requires an accurate time signature and, if there are any, identification of pick-up measures. See sections on Time signature and Bar line specification.

The metronome can beat an advanced alternating time signature (where the beat changes from one measure to the next), provided the "*Alternating Signature and Metronome*" option is chosen in the time signature.

4.2.3.2. *Metronome Channel*

In both the *Hear* and the *Record* dialog, it is possible to specify the MIDI channel and pitch for the metronome. If channels 1-16 are selected, the metronome will be played on that MIDI channel on the output device. If channel 0 is selected, the metronome will be the current system 'beep' on the computer speakers.

4.2.4. *Tracking*

If the Tracking option in the Hear option dialog is checked, a tracking line will move across the score as the piece is played. The line identifies the bar being played which, if necessary, will be scrolled into view, where possible allowing following music to be seen. While tracking, the piano window is minimized, so the music is not obscured. At the very end of the piece, the tracking bar will be moved briefly to the end of the final system.

The default state of this checkbox may be specified in Lime's general preferences. In addition there are options in the general preferences to specify the desired appearance and colour of the tracking line.

For Lime Lighter subscribers, there is an additional '*Lime Lighting*' tracking option, which tracks using a highlighting rectangle. This is similar to using the automatic scrolling pedal, except that the window will not be maximized. It can be useful for rapid testing while a piece is being set up.

4.2.5. *Skip Repeats*

If 'Skip Repeats' is selected in the Hear dialog, repeats will be ignored and any possible discontinuity will be played as though it were the last time through.

If it is partially selected, only the next repeat or other playing discontinuity will be treated as the last time through; any following ones will be honoured.

4.2.6. *Do Not Reset Midi*

By default, the controls on the MIDI output are reset to the default before playing begins (this does not affect the instruments). If checked, *Do Not Reset Midi* stops this happening.

4.2.7. *Ignore Sustain Ped*

If fully checked, all pedal annotations will be ignored. If partially checked, any *Ped* annotation will be stopped after a short period if there is no *Ped up* (*).





4.2.8. Use Piano if Voice has no Instrument

If checked, used midi channels, with no default MIDI program, will be set to 'Piano' before play-back begins. Any voice with a MIDI specification will change this, but any that do not, will play as 'Piano'. If this is not set, and a voice has no default MIDI program or MIDI annotation, the current setting of that channel will be used.

4.2.9. Hear to End of Piece

If Command-Down-Arrow (Control-down-arrow on Windows) is pressed, the piece will be played from the current position with a metronome click; if the shift key is also pressed, the metronome will be omitted. All parts in the current context will be played, regardless of anything configured in the *Hear* option.

4.2.10. Voices in Context

Both the Hear option and Midi export limit the parts that can be heard/exported to those voices that appear in the context. If you want a part to be heard but not seen, assign it to the context, but use *Parts and Voices* to assign all voices in the part not to print (*Don't Print* check-box). The *Voices and Parts...* dialog allows you to specify that the arrangement should continue to the end of the piece for non-printing parts/voice only.

4.3. Default MIDI Instruments (Programs)

The default program (instrument) for each voice may be set using *Voice->Voice on Channel*. If a default is set then, whenever the voice is played, the MIDI channel will be set to that program when hearing is started. The number is the same as used for interpreted *Program* annotations (i.e. starting from 1).

With Lime 9.16 and later, it is strongly recommended that this method of setting the MIDI instruments at the beginning of the piece is used in preference to the old interpreted *Program* annotations.

Any subsequent *Program* annotations will override this default, except that *Program 0* will revert to the default, if any.

When a default program is set in the *Voice on Channel* dialog, when the user presses *OK*, after making any changes, Lime will check it does not conflict with other voices on the same channel. If there is a conflict, the user is warned with the option to fix the first conflicting voice. This check is not done if the Shift is pressed when *OK* is clicked.

4.3.1. Piano Sounds

If a piece is active and the piano keyboard (in the piano window or on an external MIDI device) is played, the channel used will be that of the currently selected voice (1 if none) and the instrument used will be the one in place at the beginning of that voice, if any. If there is no selected voice or voice has no specified instrument, then the sound will be whatever is currently set for the channel.

Unless in note entry mode, the pitch will normally be that of the note played. In note entry mode for a transposing instrument, there is a preference option (see section 21.4.3) to use the pitch that will be heard when hearing the music. In normal mode, such transposing can be turned on temporarily by turning on CAPS-lock (shift-lock) on the computer keyboard (do not forget to turn it off when done).





4.4. Resetting MIDI

The *Hear->Reset MIDI...* menu option, resets the currently selected, output MIDI device, stopping all notes and, optionally setting the MIDI instrument to Piano (1). Optionally, this can be restricted to channel 1, but otherwise it is done for all 16 channels.

If used with the Shift key, all notes are stopped on all channels and the instrument set to Piano, but the channels are not reset (and new devices are not detected).

4.4.1. Detecting New MIDI Devices

If a new MIDI device is connected, because it is possible that it will affect current MIDI output (if it is the actual preferred device), it is necessary to reset MIDI on all channels in order for Lime to know about it and start using it. Typically one should then use *Hear->MIDI Output* and/or *MIDI Input* to select the device.

4.4.2. No Sound

All music sounds are played though the preferred MIDI output device. If the preferred device is not available, then there may be no sound. Use *Hear->MIDI Output* to set an available device.

Note that, after 9.16.97, if the preferred output device is not available, the default will be used instead (on Macs, *QuickTime* and on Windows, *MidiMapper*; however before that, silence will reign).

4.5. Playable Pitch Range

In 9.16, using *Voice->Voice on Channel*, it is possible to specify the playable concert pitch range of a voice or all voices in a part. This is used in conjunction with the preference option to '*Highlight Notes with Unplayable Pitch*'; if this is set, any notes whose play-back pitch is outside the specified range will be highlighted, usually in red. Note that this has no effect on MIDI playback, all notes will be played, regardless of the specified playable range.

Range values are specified in '*scientific designation*¹¹', with C4 being middle C (and B3 the semitone below it). Currently the valid range is C-1 through G9. If required '#' or '+' can be appended to designate sharp, and 'b' or '-' to designate flat. If one of the ranges is blank (the default), then no limit will be applied for that value.

4.6. Microtones, Pitch Bend and Portamentos

Lime can play microtonal tunings (more than just the equal-tempered scale). On most midi devices, in order to play any microtone (i.e. not an exact semitone), it is necessary to use *Pitch Bend* on an exact semitone. *Pitch Bend* affects the whole channel, so it is important that no other notes are playing on the channel at the same time. If you want microtones to be heard correctly, then you should set the synthesizer for the voice for microtonal tuning (or, if you have it, a Yamaha FB-01 or equivalent). This is done with the *Voice->Voice on Channel...* dialog:

¹¹ C4 means C, 4th octave from left of a traditional piano (hardly scientific, but a useful convention).

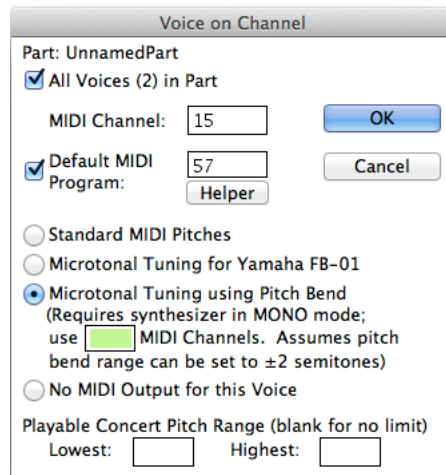




4.7. Modify Playback and Re-compute Playback

Any change made to playback using the *Hear->Modify Playback* menu will be rescinded if playback is recomputed for any reason.

If a permanent pitch change is required then one should use key signature transposing or octave shift; if necessary the key signature can be hidden (see 15.5).



The dialog box is titled "Voice on Channel". It contains the following elements:

- Part: UnnamedPart
- ☒ All Voices (2) in Part
- MIDI Channel: OK
- ☒ Default MIDI Program: Cancel
-
- ☐ Standard MIDI Pitches
- ☐ Microtonal Tuning for Yamaha FB-01
- ☒ Microtonal Tuning using Pitch Bend
(Requires synthesizer in MONO mode; use MIDI Channels. Assumes pitch bend range can be set to ± 2 semitones)
- ☐ No MIDI Output for this Voice
- Playable Concert Pitch Range (blank for no limit)
Lowest: Highest:

You can use the microtonal tuning option provided you can set the range for pitch bend to ± 2 semitones (which is the normal default for most synthesizers). Tuning is accurate to 0.5 cents (0.005 of a semitone). If polyphony or more than 1 voice is needed, then the number of midi channels should be set to the number of simultaneous notes (minimum 1). The additional channels used are those above the base midi channel.

Similarly pitch bent slurs (portamentos) or ties require just one note to be playing (unless all are bent together). The maximum pitch bend is ± 12 semitones (i.e. an octave).





5. Short Cuts

5.1. Menu Short-Cuts

When upgrading from a version of Lime earlier than 9.15, it is highly likely that the menu short cuts are no longer correct. Use the '*Edit->Preferences->Shortcut Keys...*' menu option to reset them to the default (click the relevant check-box) and then, if you need to, change them.

New built-in short cuts include:

cmd/ctrl-.	Invoke Hear option or suspend hearing.
cmd/ctrl-,	Invoke annotation placement and justification dialog.
cmd/ctrl-;	If there is no open piece at all, invokes the <i>New Piece</i> dialog; otherwise Toggles between previously viewed context and the current one. With shift key, goes to Score context or, if already there, to the last context in the list.
cmd/ctrl-'	Toggles special accidental piano window mode (single quote).
cmd/ctrl-[Normally, go to previous page; but if play-back with tracking is active, play slower.
cmd/ctrl-]	Normally, go to next page; but if play-back with tracking is active, play faster.
cmd/ctrl-shift-[Go to first measure of first page.
cmd/ctrl-shift-]	Go to last measure of last page.
cmd/ctrl-alt-[or]	Go to page dialog (n.b. on Windows, Alt cannot be AltGr).
cmd/ctrl-shift-space	Go to the last note that was selected before the current one.

Use of the shift-key with any context switch will open the context in a new view.

5.2. Changed Default Short Cuts

cmd/ctrl-G	Go to Bar dialog (default used to be 'Go to Page'). The resultant defaults to 'Go to Bar', but may be used to 'Go to Page'.
cmd/ctrl-shift-G	Go to 1 st bar of current page. G is whatever letter (if any!) is configured as the ' <i>Go to Bar</i> ' short cut.

Note that on Macintoshes, cmd-H is no longer defaulted to invoke the hear option (use cmd-period instead). This is to avoid the conflict with the standard use of cmd-H for hiding the application.

5.3. User Defined Short Cuts

The built in short cuts (i.e. ctrl/cmd-punctuation) are permanent and cannot be changed by the user. However you may, if you wish, configure an additional alphabetic short cut; if you do, the build-in one remains available.

On both Windows and Macs, the set of short cuts, **ctrl/cmd-alt-letter** may be configured by the user. Lime will never pre-assign any of these. On Macs, the facility to assign short cuts to ctrl-letter has been replaced by cmd-alt-letter (*ctrl-letter is deprecated*).

On Windows, the alt-graphics (AltGr) key cannot be used for this purpose.





5.4. TAB and Number Pad Arrow Key Short Cuts

When not editing annotations or within a dialog, the **TAB** key navigates from bar to bar, changing pages if necessary. Gathered bars of multi-measure rests are treated as one bar. By itself, **TAB** moves forwards to the chord after the next, or with **Shift** back to the previous, bar line. All notes on the selected staff will be played.

If your keyboard has arrow keys and/or a number pad, which is not locked in 'Num-Lock' mode¹², a number of short cuts are available (*Ctl* means the Command key on Macintoshes and the Ctrl key on Windows). Note that individual notes or chords are only played if the preference option 'Play sound when *Navigating with Arrow Keys*' is enabled.

HOME_CHAR (num pad 7)

- by itself move back one bar and play all notes on staff
- +ctl move to previous chord and play all notes in time-slice

UPARROW_CHAR (num pad 8)

- by itself or +ctl+alt go to previous voice and play note
- +ctl play all notes in current time-slice

PAGEUP_CHAR (num pad 9)

- by itself move forward one bar and play all notes on staff
- +ctl move to next chord and play all notes in time-slice

LEFTARROW_CHAR (num pad 4)

- by itself or +ctl+alt go to previous note and play note
- +alt go to previous annotation
- +ctl go to previous chord and play all notes on staff
- +shift group select previous note
- +shift+ctl group select previous voice-chord

NUMPAD5_CHAR

- by itself play current note (*and, for Lime Aloud users, speak all details*)
- +ctl play all notes at current time on staff

RIGHTARROW_CHAR (num pad 6)

- by itself or +ctl+alt go to next note and play note
- +alt go to next annotation
- +ctl go to next chord and play all notes on staff
- +shift group select next note
- +shift+ctl group select next voice-chord

END_CHAR (num pad 1). *For Lime Lighter users only - emulates the pedal.*

- by itself *move to and highlight next measure*
- +shift *move to and highlight previous measure*
- +ctl *take-repeat and highlight measure at its beginning*

DOWNARROW_CHAR (num pad 2)

- by itself or +ctl+alt go to next voice and play note
- +ctl play from current selection to end of piece with (+shift without) metronome

¹² On some Windows keyboards, notably the latest from Dell, the number pad is always in 'Num-Lock' mode and the Shift- or Ctrl- cannot be used with them.





6. Pre (and post) 9.15 Scores

In general, when reading scores created with earlier versions, Lime will automatically update the score to the new format. Since 9.14, Lime has always been capable of reading the next file format, albeit with possible loss of some features.

6.1. Upgrading Issues

There are a number of issues associated with upgrading scores created before Lime 9.15, particularly those created before 9.14.

As these can result in appearance changes, it is recommended that the results be checked, particularly if it is intended to publish them in some shape, size or form.

6.1.1. Annotations on Invisible Rests

There was a problem in earlier versions of Lime that caused annotations on some invisible rests not to be shown (on multi-voice staves many rests are made invisible because they would conflict with notes or rests on other voices). This was fixed in Lime 9.15 and now annotations are shown, whether or not the associated rest is visible. This is particularly important for handling scores imported via MusicXML.

When upgrading older pieces, Lime attempts to identify rests that would have been invisible and automatically hides any associated annotations. It can only do this if there are no other contexts in which the voice staff assignment is different. It is, therefore, possible that previously invisible annotations will appear after conversion.

Some users will have added additional annotations when an annotation did not appear in a context, particularly for annotations declared as '*duplicate in all contexts*'. Lime attempts to identify such additional annotations and hide all but one. If the additional annotations were not in exactly the same place, it can appear that an annotation has moved after the upgrade.

6.1.2. Invisible Rests become Visible

An old, related bug was that on a multi-voice staff, if at some point one voice had a normal, unhidden rest, but one or more of the others had hidden rests, the unhidden rest would often (but not always) become invisible. Lime 15, correctly, ensures that the unhidden rest is visible. This, of course, means that not only can a rest appear where it wasn't before, but also any associated annotations become visible.

6.1.3. Stem Directions

After Lime 9.05, the algorithms for determining stem directions and associated beaming on multi-voice staves are considerably more sophisticated (improved) where stem directions have not be explicitly set. This can, of course, cause a change in the appearance of scores after upgrade (usually for the better). If the old appearance is essential then, currently, it will be necessary to set stem directions explicitly.

6.1.4. Annotation Horizontal Placement

As there may be small changes in note head horizontal positioning, as 9.15 uses more sophisticated rounding algorithms, annotations anchored on note heads may move slightly.





6.1.5. Page Width

Versions of Lime before 9.14 rounded the page width up to a 16-pixel boundary (because it was needed on the original Macs). This is no longer done; so when upgrading older pieces, the page width is rounded up to compensate and so ensure there are no positioning differences after the upgrade.

6.1.6. Tuplet Plotting

If specified in the options, Lime will plot tuplet numbers whether or not the tuplet is complete. The piece option to plot tuplets may be partially set to specify that incomplete tuplet numbers are not plotted, as was the case prior to 9.15. This is set by default when upgrading earlier pieces.

6.1.7. Mark and Accent Placement

In Lime 9.05 and earlier, at any point where there were notes with different stem directions on the same staff, marks and accents would always be plotted on the stems to avoid any clash. After 9.05, this does not happen if the only notes in the other direction are invisible (e.g. invisible rests). There is a piece option in Lime to revert to the old 9.05 behaviour; this is set by default when upgrading 9.05 or earlier pieces.

6.1.8. Old Scores between Macs and Windows

Prior to 9.15 there were issues with moving scores between Macintoshes and Windows:

- The piece colour palette was not properly converted, which will affect scores with coloured notes or annotations
- Parameters that specified a character (e.g. a note head) in one of the music fonts (Marl, Tufa or Sonata) were incorrect.

Both these issues have been fixed; any scores created by the release version of 9.15 or later should be completely transferable between Macintoshes and Windows.

However, to handle older scores, it is necessary to know which system a score was created on. There is, therefore, an option when opening such a score (or within the *Batch...* facility) of specifying the provenance of a score, if it is known.

If you know definitely on which system a pre-9.15 score was created, you should click the "Known source?" and selected the required system. If the source is not known, leave this unchecked - Lime will attempt to derive it as far as possible; by default it assumes the system you are running on.

In general this is not a problem except if some Modified Staff Notation changes have been made to note heads, etc.

6.1.9. Fonts on Macs

On Macs, *Arial* (which supports Unicode) is used instead of the predefined *Geneva* font, and *Times New Roman* is used instead of *Times*. These changes should have minimal visual impact on a score, whilst ensuring total compatibility with Windows. If required, and it is installed, *Geneva* or *Times* may be used as ad hoc fonts.





6.2. 9.17 Format

Lime 9.17 uses a slightly enhanced file format. This is partly to support new features and partly for full Unicode (actually UTF-8) for annotations and for part and context names. Lime 9.16 (and 9.15) can read 9.17 files but there may be some information loss (particularly if you are still using 9.15).

In addition to full support for unicode, the additional features in the 9.17 file format include:

- Annotations anchored to the top or bottom of a system.
- Annotation positions scalable to staff/system separation.
- Vertically growable line/curve annotations.
- Note by note fingering.
- Auto-generated annotations.
- 8^{va} and 8^{vb} Clefs.
- Semi-standard quarter tone accidentals (independent of the special accidentals).
- Parenthesized accidentals.
- Key signature changes with naturals shown before bar lines.
- Some control over whether the end of a slur is near the note-head or stem of a chord.
- SMuFL (Standard Music Font Layout) font identifier.

All 9.16 versions recognize unicode annotations (actually all versions after 9.15.8). Annotations in files created by Lime 9.16 may be encoded using Unicode. However, by default, this is only done if it is not possible to render the text using the local 8-bit character set; though there is a piece option to use Unicode always. In 9.17, use of unicode is the default and will probably not be optional.

The final 9.16 re-release (9.16.99.) recognizes all initial 9.17 features. It is, therefore, recommended that all users install the re-release, while waiting for the 9.17 official release.





7. Import and Export

7.1. MusicXML Importing

LIME provides basic support for importing MusicXML. It is perfectly capable of importing any version of MusicXML (3.1 was the latest at the time of writing); however it will ignore a lot of features from MusicXML later than 1.1. Furthermore, many features from 1.1 are also ignored. In some cases this is because it is not appropriate, such as the positioning of notes or width of measures. In other cases, it is because LIME does not have an appropriate feature.

It must be recognized that, whilst MusicXML is very useful, the standard is, unfortunately, not rigorously defined (unlike MIDI); so different programs interpret it differently. Most programs, including LIME, do not have all the features implied therein and most programs (including LIME) also have features that cannot be exported with MusicXML.

Whenever a piece is imported from MusicXML, it will usually be necessary to make some manual adjustments, *particularly complex piano parts*.

Main features not supported by LIME include:

- Compressed MusicXML (.mxl files). However, if you change the suffix to .zip and use a zip tool to unzip it, the required .xml file is contained therein.
- <score-timewise> is not supported. LIME only supports <score-partwise> MusicXML.
- The smallest note duration supported by Lime is 1/128th, and the largest tuplet is 254/60. Any piece with smaller durations or crazy tuplets will not import properly and will have timing errors.

7.1.1. Unicode Support

After 9.16, Unicode and, therefore, annotations in languages not based on the Roman alphabet are supported. Your system must, of course, have the necessary fonts used for these annotations, in order to be able to render the characters.

7.1.2. Creator Funnies

Ultimately MusicXML import can only be as good as the original encoding. Except in the simplest of scores, it is actually rare, if ever, that an exported file can be imported totally accurately, even into the program that originally exported it. There are many known oddities, including:

- Sibelius (native 7) exported files have many of the credits at the bottom of the page (even when imported into itself)! It appears that it has erroneously used the distance from the top staff rather than the page bottom (as defined by MusicXML). The option to 'Discard Unclassifiable Credits' tries to detect and rectify this.
- MusicXML specifies that the horizontal positioning information for many directives is relative to the previous bar line, or system start (the other directives are relative to the note). However there seems to be no consistency, even within the same file; in many cases, it is clearly relative to the page start. Compensating for this is very difficult and, at the moment, apart from detecting Sibelius (native 7.1 and earlier), nothing much is done automatically apart from having import options. If annotations are clearly misplaced, experiment with the options.
- Creators, such as Sibelius (both native and Dolet) are erratic in how 8^{va} and 8^{vb} are placed. Sometimes they specify the notes as they are meant to be played, but often in the notated position. There is an import option to decide what to do.





- Photo Score seems to use relative-xy (instead of default-xy) attributes to specify the actual position of directives. There's an import option to correct this.
- SharpEye separates voices on multi-voice staves into different staves. Manual effort is needed to merge them again, but this is rarely difficult. Note that, when exporting from SharpEye you must use the XML 1.1 option to get reasonable annotation placement.

If you have problems with importing a piece from SharpEye via MusicXML, try using NIFF; it is *sometimes* better with multi-voice staves. However, MusicXML is recommended.

7.1.3. Notation Ornaments

If multiple ornaments are specified on the same note, Lime uses the "*Stack Vertically*" style to place them one above the other. It should be noted that "*Stack Vertically*" is also used on single delayed turns, which are built using more than one character.

7.1.4. Fingering and String Specification

MusicXML allows fingering to be specified for each note. Lime treats fingering as a stacked annotation applying to the whole chord. Each fingering specification will be *Stacked Vertically*. A change of finger (substitution) is represented by a dash (-) and an alternative is represented by a slash (/).

A string specification is included in the fingering and is enclosed in parentheses.

7.1.5. Dynamics and Micro-Tones

As of 9.16.2, notes' dynamics (volume) and any microtone pitch values are recognised. However, unusual accidentals are not yet handled. If a piece is known to have micro-tones and special accidentals, the preference option '*Identify Notes with Modified Pitch*' can be used to highlight such notes so that their accidentals can be set by hand. It is important NOT to re-compute play-back until all such accidentals have been set.

7.1.6. Tuplets

Lime currently (9.16.9) only supports one level of tuplet nesting. If the MusicXML specifies more than one level, the sub-levels will be combined into a single composite tuplet.

Tuplet brackets are only imported for the first tuplet level.

7.1.7. Slurring

Lime does not support implicit slurs between staves. This means that, in complex piano parts, where voices change staff, there will very likely be anomalies in the slurring, which will need manual correction. Furthermore, MusicXML does not distinguish between a real slur and a curve.

7.1.8. Tempo Marking

The MusicXML `<metronome>` elements are imported as tempo marking annotations. In a multi-part piece, these annotations are preset to *Duplicate in Every Part context* (see section 3.19.1). This means that, after importing, if contexts are created with just some parts, the tempo marks will still be printed.

As Lime does not support additive signatures with different denominators, these are adjusted to the lowest common multiplier of the denominators.

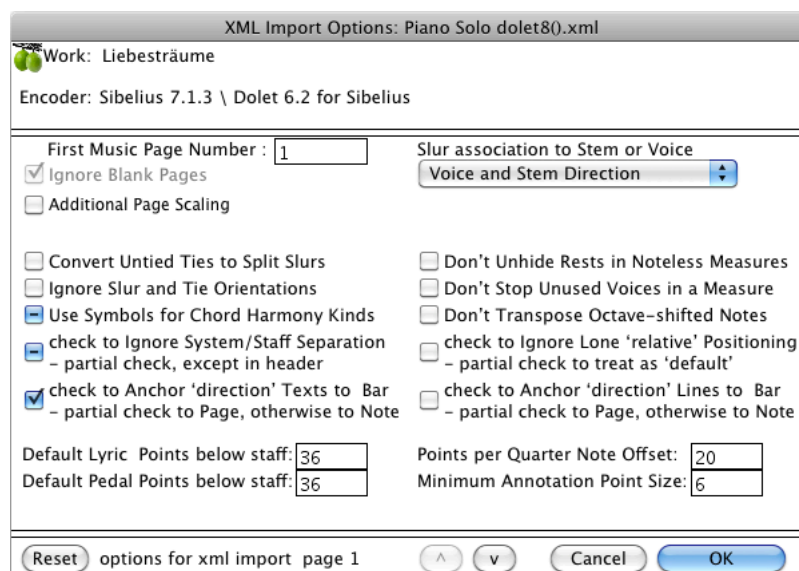




7.1.9. Importing Options

Whenever a MusicXML file is imported, the user is presented with a dialog to specify the options for the import. This is a two-page dialog, with advanced features on the 2nd page. At the top of the dialog are meta-data extracted from the MusicXML header, identifying the encoding source and the work title (though many encoders omit this and it will be blank).

Some of the options are 3-way. A hyphen on a Mac or a dimmed tick on Windows indicates the third alternative.



XML Import Options: Piano Solo dolet8().xml

Work: Liebesträume

Encoder: Sibelius 7.1.3 \ Dolet 6.2 for Sibelius

First Music Page Number : 1

☒ Ignore Blank Pages

☐ Additional Page Scaling

☐ Convert Untied Ties to Split Slurs

☐ Ignore Slur and Tie Orientations

☒ Use Symbols for Chord Harmony Kinds

☒ check to Ignore System/Staff Separation - partial check, except in header

☒ check to Anchor 'direction' Texts to Bar - partial check to Page, otherwise to Note

☐ Slur association to Stem or Voice

Voice and Stem Direction

☐ Don't Unhide Rests in Noteless Measures

☐ Don't Stop Unused Voices in a Measure

☐ Don't Transpose Octave-shifted Notes

☐ check to Ignore Lone 'relative' Positioning - partial check to treat as 'default'

☐ check to Anchor 'direction' Lines to Bar - partial check to Page, otherwise to Note

Default Lyric Points below staff: 36

Default Pedal Points below staff: 36

Points per Quarter Note Offset: 20

Minimum Annotation Point Size: 6

Reset options for xml import page 1

Cancel OK

MusicXML Import Options Basic Options

- **Additional Page Scaling**
If checked, the music is scaled up a little bit to compensate for the fact that some programs have slightly smaller note-heads, etc than Lime. If partially checked, you can specify the scaling.
- **Convert Untied Ties to Split Slurs**
If the source seems to use a tie as a slur (often true of Sibelius), convert it to a Slur. Otherwise leave it as a pseudo tie, which have been supported since Lime 9.15.
- **Ignore Slur and Tie Orientations**
If checked, then the curve direction of all slurs and ties will be assigned by Lime; all orientation indications in the XML will be ignored. If several slurs and ties seem to be stupidly placed, then checking this might be the solution.
- **Use Symbols for Chord Harmony Kinds**
If the MusicXML contains chord symbols (<harmony> elements), this provides the option to use symbols, rather than words for certain features. The default, partially checked, means only if specified in the MusicXML. If fully checked, symbols will always be used where possible and, if not checked (recommended for Lime Aloud or GoodFeel users), they will never be used.
- **Ignore System/Staff Separation**
If checked, the source's system and staff separation values will be ignored. If partially checked (the default), those in the header will be honoured, but not those in individual parts. If you want the score to be laid out as close as possible to the original, then this should be unchecked, but it can result in a lot of parameter annotations to set the separations.





- Don't Unhide Rests in Noteless Measures
Normally, if a measure on a staff contains nothing but rests in all voices, all rests are made not hidden. In most cases the formatter will then combine them into whole bar rests. This tries to ensure that no measure is totally empty. Check this option if empty measures are actually required.
- Don't Stop Unused Voices in a Measure
Normally, if an imported voice in a multi-voice system has only rests and no annotations in a measure, Lime will stop printing it until it becomes active again (if ever). Occasionally, this can cause problems with (low Vision) tracking.

This option causes Lime to hide the rests, rather than stopping printing the voice; but it can cause difficulty with part extraction and other problems such as spacing and prematurely terminated beams, so it is not recommended except to cater for poor XML files.
- Don't Transpose Octave-shifted Notes
Normally when 8^{va} or 8^{vb} is specified, the MusicXML specifies the notes at the pitch they should be played. These need to be transposed for notation purposes. This option supports encoders that specify the notes as they should be notated, and turns off the transposition.
- Ignore Lone relative-xy Positioning
If checked, relative positioning (relative-x and relative-y) are ignored if there is no corresponding default positioning attribute (default-x or default-y). If partially checked they will be treated as the corresponding default attribute and horizontally (x) always relative to the note head. Try this if annotations are consistently badly positioned, particularly vertically.
- Text and Line Anchoring
Although MusicXML specifies that the position of directions is relative to the previous bar line (or beginning of system), this is often not adhered to, even in the same file, particularly for Lines (Hairpins, etc). If many Text or Line annotations seem to be misplaced, try changing one of these options. Sibelius 7.1.1 and earlier, for example, placed many text annotations relative to the page (partial check the option); this was fixed in Sibelius 7.1.2.
- Default Lyric and Pedal Points
If the position of Lyrics or Pedals are not given in the MusicXML (usually the case), place them the specified number of points *below* the staff. The only way of getting the correct value is to experiment. Of course, if there are both lyrics and pedal marks on the same staff, the values will need to be different.
- Points per Quarter Note Offset
This is a kludge to cater for MusicXML files which specifies the horizontal position of annotations, particularly hairpins ("wedges") in terms of musical position, which is not supported by lime. This option specifies an approximate number of points per quarter note (crotchet).
- Minimum Annotation Point Size
This imposes a minimum size for text annotations. Use this option if annotations seem stupidly small (a minimum size of 12 would probably be appropriate).
- Slur Association to Stem or Voice
Specifies how slurs on staves with multiple voices are handled by default in the whole piece; it may be changed later using the *File->Options for Context* menu. For most XML, the default, 'Voice and Stem Direction' seems to give the best results. Note that Lime's normal default is 'Stem Direction' only.



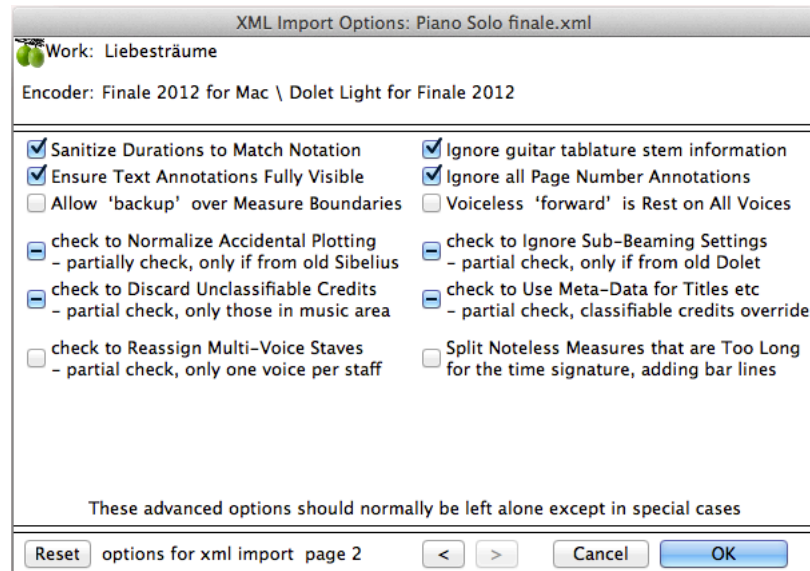


The ☒ button shows advanced options on page 2. In most cases, the defaults are appropriate and will not need to be changed.

- Reassign Voices on Multi Voice Staves

If this is checked, Lime will attempt to split each staff into two voices based on the designated stem direction of each note. If it is partially checked there will be just one voice per staff. If it is unchecked (default), then the original voices will be used.

Note that voice reassignment is not possible if the piece has a voice that hops from one staff to another. If such a voice is detected, reassignment will be abandoned and the original voices used.



XML Import Options: Piano Solo finale.xml

Work: Liebesträume

Encoder: Finale 2012 for Mac \ Dolet Light for Finale 2012

<input checked="" type="checkbox"/> Sanitize Durations to Match Notation	<input checked="" type="checkbox"/> Ignore guitar tablature stem information
<input checked="" type="checkbox"/> Ensure Text Annotations Fully Visible	<input checked="" type="checkbox"/> Ignore all Page Number Annotations
<input type="checkbox"/> Allow 'backup' over Measure Boundaries	<input type="checkbox"/> Voiceless 'forward' is Rest on All Voices
<input type="checkbox"/> check to Normalize Accidental Plotting - partially check, only if from old Sibelius	<input type="checkbox"/> check to Ignore Sub-Beaming Settings - partial check, only if from old Dolet
<input type="checkbox"/> check to Discard Unclassifiable Credits - partial check, only those in music area	<input type="checkbox"/> check to Use Meta-Data for Titles etc - partial check, classifiable credits override
<input type="checkbox"/> check to Reassign Multi-Voice Staves - partial check, only one voice per staff	<input type="checkbox"/> Split Noteless Measures that are Too Long for the time signature, adding bar lines

These advanced options should normally be left alone except in special cases

Reset options for xml import page 2 < > Cancel OK

MusicXML Import Options Advanced Options

Describe the remaining advanced (page 2) MusicXML import options.

7.2. MusicXML Exporting

Although Lime can export MusicXML, the result is currently relatively primitive and is limited to basic information. This will be improved in the future, but for the moment development effort is being concentrated on MusicXML import. As of 9.16.2, Note dynamics and microtones are exported.

7.3. MusicXML File Name Extensions

Files ending in .xml and .musicxml are treated by Lime as MusicXML files.





7.4. NIFF Import

Whilst Lime can import NIFF from any provenance, it is strongly recommended that, with the possible exception of inter-working with *SharpEye*, MusicXML is used. NIFF import is essentially deprecated.

It is important that, before creating a NIFF file (e.g. from SharpEye) for importing into Lime, all rhythm and note duration errors have been corrected.

Note that the facility (in Lime 9.0n) for preserving NIFF graphical alignment and flagging possible errors with coloured ("red") rests is no longer supported, because it had too many problems to be generally useful.

7.4.1. SharpEye

It should be noted that *SharpEye* is a very old program that has been superseded by *PhotoScore* (www.neuratron.com/photoscore.htm), which has further developed the SharpEye scanning engine. There are also many other, more modern, music scanning engines¹³.

NIFF occasionally gives better results when importing scanned files from SharpEye when there are multi-voice staves. This is because, with MusicXML, SharpEye separates the voices onto different staves, which need to be merged again, but this is rarely difficult. However there are very many cases when MusicXML is better. Therefore, only if importing using MusicXML has problems, try NIFF.

N.b. When saving XML files from SharpEye, it is important that the XML 1.1 option is set.

7.5. NIFF Export

Support for NIFF export has been discontinued¹⁴.

7.6. Scanned Music

It must be recognized that creating pieces by optical recognition of a scanned printed score is prone to error, however good the scanning utility is. Unless the scan is reasonably clear and not fuzzy, the results can be unusable. The recommended scanning quality is 300dpi, and 256 shades of grey, often referred to as "Black and White Photograph"; simple 2-colour black and white is generally unusable.

It is important that most rhythm errors are fixed in the scanning utility before exporting to Lime. The ability to see the results compared to the original is pretty essential.

Lime has a number of facilities to assist in fixing errors in scanned pieces, see section 8.

¹³ Including SmartScore (www.musitek.com)

¹⁴ It is far more important to support exporting MusicXML.





8. Duration Error Fixing

If the piece has been created using Lime, duration errors (rhythm and timing) are fairly rare and are generally caused by user error or deliberately fiddling about. However, if the piece is imported from MusicXML or NIFF, such errors are possible, particularly if it is a scanned piece. Even if it is not scanned, the ambiguities of MusicXML, particularly with multi-voice staves and when voices move between staves, can cause timing errors, which need to be corrected.

Rhythm errors in a piece can be manifested in a number of different ways, including:

- Bar lines on different staves do not align and the context is not polymetric¹⁵.
- At places in a non-polymetric context, voices have different numbers of bar lines.
- At places in the piece, voices have different numbers of beats.
- Inconsistent or incorrect playback.

Most of these are reported by Lime either when a piece is opened or as the user navigates through it.

The main tools for fixing duration errors are:

Fixer Context ¹⁶	<p>It is possible to set up a special <i>fixer</i> context to facilitate duration error correction. A fixer context is designed to make it easier to identify where duration errors are. When first set up all voices are configured to be printed on separate staves and, furthermore, all rests are shown, hidden or otherwise.</p> <p>In addition a fixer context allows insertion and deletion of measures, which is otherwise only available in the score context; this is because a fixer context always contains all voices (like the score).</p>
Enabling Duration Edits	When duration edits are enabled, the length of a note can be changed without any time signature constraints. Notes and rests can simply be deleted (killed). Furthermore, additional rests can be inserted before or after a chord.
Finding Duration Error	The <i>Page>Fix Problems</i> sub-menu offers facilities to find the <i>Next Rhythm</i> error after the currently selected note. This can either be on the same voice or on any voice.

8.1. Fixer Contexts

A *Fixer Contexts* is a temporary context designed to make it easier to identify where duration errors are and, therefore to fix them. A fixer context can be set up quickly using the menu item:

Page>Fix Problems->Context for Duration Fixing...

It can also be set up using Part Extraction as an option when duplicating the score.

Once a fixer context has been set up, the *Context for Duration Fixing* menu item will switch to it and, if invoked with the Shift key, will give the option to reset all the defaults.

¹⁵ If the context is polymetric (see section 17), bars line will probably not align, but in all other cases misalignment is an indication of rhythm errors.

¹⁶ Fixer contexts are available from 9.16.9.





Very often on multi-voice staves, the duration error is a missing rest or a spurious one, which is not seen. In a fixer context, all notes and all rests are printed and should, therefore, be identifiable.

The main features of a fixer context are:

- It comprises all voices which are initially set up on separate staves, such that all notes and all rests are printed. Many errors are obvious because of bar line misalignment (though, of course, you need to be able to see that).
- A fixer context allows insertion and deletion of measures, which is otherwise only allowed on the score context.
- All hidden symbols are always shown (this cannot be turned off) and rests are never combined.
- Single bar lines are slightly thicker than normal to make misalignments easier to see, when they do not go between staves.
- All bar lines are extended to the next staff in a system. They are drawn dashed when they are not specified as between staves.

Apart from the few special formatting features, a fixer context is just like any other context. However whilst it can be printed, it is unlikely to be suitable for ordinary use. When a fixer context is first set up, as well as separating all voices onto separate staves, the following are pre-set in the fixer context:

- All voices are on separate staves and are always shown. There are no staff changes, but any stemming between staves is retained.
- System braces are used to identify staves that belong to voices in the same part.
- Bar lines between staves at the end of systems match the system braces.

These settings are designed to assist in identifying discrepancies. They can, of course, be changed, should you so wish.

8.1.1. Voices in a Fixer Context

Fixer contexts always include all voices; a voice cannot be removed from a fixer context. If you do not want to see a voice, you can use Parts and Voices to specify it as not printing. For example, if there are many voices that are known to have no problems, stopping all but one or two of them might be desirable.

8.1.2. Ticker-Tape

In most cases, Ticker-Tape can be useful for a fixer context, because it allows one to see bar-line misalignment over system breaks. A ticker-tape (panorama) context page always has just 3 systems concatenated into one long system (known as a page portal); see section 20 for more information on ticker-tape contexts. If it is a fixer context, bar lines at original system breaks are shown in red, with the bar line number above the top staff. Bar lines at original system breaks are placed independently on each voice and therefore, if there are duration discrepancies, they will be misaligned.

Ticker-Tape is the default for a fixer context if created using the Fix Problems menu option, *Page->Fix Problems->Context for Duration Fixing...*

8.1.3. Oddities

Do NOT use a fixer context to format the music. It must be recognized that, because of the voice separation, many things may be a little strange in a fixer context, particularly if there were multi-voice staves in the score or if voices moved between staves. Such oddities can include strange beaming or slurring and incorrect alignment of annotations.





8.1.4. Saving

If the piece is saved while in a Fixer Context, it is saved such that, when opened again, it will be in the Score Context. This reduces the chances of inadvertently formatting a fixer context.

8.1.5. Redundancy

Once duration errors have been fixed, unless you are in the habit of creating new ones, the fixer context should be deleted, using *Context->Part Extraction*.

8.2. Duration Editing



Extract of a Fixer Ticker-Tape Context from a scanned piece, showing many rhythm and structure errors.

Using *Edit->Enable Duration Edits* or *Page->Fix Problems->Enable Duration Edits*, features are enabled that allow:

- The length of a note to be changed without any time signature constraints.
- Notes and rests can simply be deleted (**killed**).
- Additional rests can be inserted before or after a chord.

When duration editing is enabled, the piano window shows the special buttons in **red**. In general, when a note is selected, and not in note entry mode, clicking on a note type, or using the keyboard equivalent, will cause the length of the selected notes to be changed to the specified value. This also applies to tuplet values. This is mostly described in the old Lime manual.

Also, when duration editing is enabled, if the Piano Rest symbol is clicked (or the keyboard equivalents, **R** or **I**), a new rest will be inserted before or after the selected note and become selected. This rest will initially be the same duration as the selection, but can then be changed using the other duration editing facilities. If Shift is used, the new rest will be placed after the selected chord, otherwise it is placed before it.

Duration editing should be disabled once it is finished with.





8.3. Finding Duration Errors

The *Page>Fix Problems* sub-menu offers facilities to find the *Next Rhythm* error after the currently selected note. This can either be on the same voice or on any voice. For this purpose a rhythm error is one where the length of a measure does not match the time signature. This is fairly primitive and may include pick-up measures, which you should, of course, ignore.





9. Colours

9.1. Music Window Appearance

There are preference options specifying the basic colouring to display the music window on the screen. The options are:

- **Lime Default.** This is black music on an off-white, cream background designed to reduce glare.
- **Black on Yellowy** Specifically designed for vision-impaired accessibility, this is normal black music on a yellowish background.
- **Black on White** (on Macs)
- **Windows Default** (on Windows). This uses the system default. On Windows this is based on the theme or, prior to Windows 8¹⁷ as set in the personalization control panel. On Macs it is always simply black on White.
- **White on Black** (with colour inversion). Also designed for vision-impaired accessibility, this displays the music as white music on a black background; **all colours are also inverted (so, for example, yellow will appear as blue).**

Whatever preference is selected, it only affects the display; printing is unaffected and will always be the standard black on white (or whatever colour paper is actually used).

9.2. Colour Palettes

There are two¹⁸ colour palettes:

- **Piece Options palette** used for colouring of annotations and of selected note-heads (including rests) in the piece. Each piece has its own palette, which is saved as part of its general options.
- **Preferences palette** used for local functions, including the automatic colouring of note heads by note name and the colour of the tracking bar when hearing and Lime Lighter highlighting.

The default colours for the colour palettes are the 16 (excluding White) standard colours used by web applications, which have standard names - black, red, lime, blue, yellow, magenta (or fuchsia), cyan (or aqua), teal, silver, olive, purple, orange, green, navy, maroon, and grey.

1 black	2 red	3 lime	4 blue	5 yellow	6 magenta	7 cyan	8 teal
9 silver	10 olive	11 purple	12 orange	13 green	14 navy	15 maroon	16 gray

Standard Colours

¹⁷ Note that on Windows 8 and later it is no longer easy to specify colouring of things like background of individual windows (though it can be done in the registry).

¹⁸ Prior to 9.16, there was only the *piece options palette*.



9.2.1. Customisation

In all colour dialogs, it is possible to customise any of colours (except black) in the associated palette. This is done by shift-clicking the colour and using the resulting dialog to set it as required.

As well as customizing an individual colour, you can reset all the colours in the palette to the default. If a colour has been customized, then no name will be displayed on its button, though behind the scenes, the button will be labelled with its red-green-blue (RGB) values.

9.2.2. Inverted Colours

If the “*White on Black (with colour inversion)*” music window appearance option is used, then **all colours are also inverted (so, for example, red will appear as cyan)**. It is important to take this into account when choosing colours for display purposes.

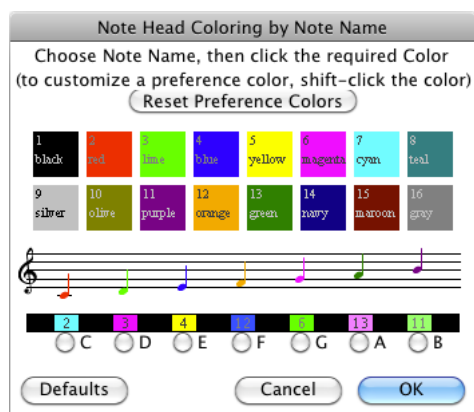
Note, however, that when the piece is printed, the result will be black on white, with normal colouring.

9.3. Colouring Note Heads and Keys by Note Name

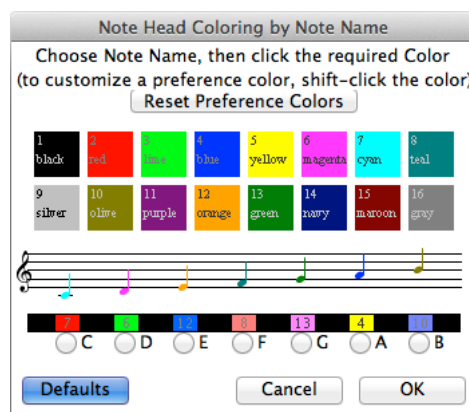
Using the preferences, it is possible to specify that every note and key signature throughout every piece is displayed in a colour based on the note name. In the preferences dialog, if the “*Color Note Heads and Keys by Note Name*” check-box is checked, it means:

- **Fully Checked** All Note heads are coloured by note name, and the colouring overrides any piece based colouring of individual notes.
- **Partially Checked** Uncoloured note heads are coloured by note name, but not those that have piece based colouring.

The mapping can be changed in the general preferences using the **Colors** button beside the “*Color Note Heads and Keys by Note Name*” checkbox.



Normal Defaults



Inversion Defaults

Note Head Colouring Dialog

To set the colour for a note, click on the note (or its radio button),



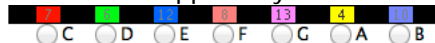
then click on the required colour button.



If the music window appearance were "*White on Black (with colour inversion)*" then, as explained above, all colours will be inverted and the normal defaults would become:



For this situation, if the music window appearance is "*White on Black inverted*", when the dialog is invoked, an alternative set of defaults will be applied by the "*Defaults*" button:



This is the best approximation to the normal defaults that can be achieved without modifying any of the colours. Note that, in "*inverted*" mode, the actual colour buttons (and displayed notes) are the uninverted colours; you will need to experiment to see how they look when inverted (as shown on the big black line).

9.3.1. Default Note Name Colour Mapping

As shown above, the default mapping to the 16 colours in the colour palette is:

Note name	Default Uninverted		Default Inverted	
	Index	Colour Button	Index	Colour Button
C	2	Red	7	<i>inverted cyan</i>
D	3	Lime green	6	<i>inverted magenta</i>
E	4	Blue	12	<i>inverted orange</i>
F	12	Orange	8	<i>inverted teal</i>
G	6	Magenta	13	<i>inverted green</i>
A	13	Green	4	<i>inverted blue</i>
B	11	Purple	10	<i>inverted olive</i>

Remember that, to set the default inverted colours, ensure the "*White on Black (with colour inversion)*" option is set in the preferences before invocation of the note head colouring dialog; then click the Defaults button in the dialog.

Note that, if any if the selected preference colours have been customized then, of course, the mapping between index and the actual colour will be different.

9.4. Colouring Issues

It is recognised that Lime's colour facilities are a little antiquated and are due for overhaul after 9.17. In the meantime, the following should be noted:

- Only the colours in one of the two colour palettes can be used, though any of the colours, except black, can be customized.
- As described above, if the music window appearance is "*White on Black (with colour inversion)*", **all** colours are inverted. The results may not always be desirable.
- Prior to 9.15.9, the prevailing preferences palette was always used for display purposes and saved with a piece. 9.15.9 now uses and retains separate palettes saved with each piece, which are independent of the preferences palette.





10. System and Staff Distance Layout Options

The default distances between systems and staves may be set using the Layout (File->Layout) dialog. The defaults are as follows:

- *Default Points between Systems* 80.
- *Default Extra Points for Separators* 30.
- *Default points between staves in a System* 80.

The layout option values can be supplemented with the new additional separation parameters (see below).

In the layout dialog, if the "separators" checkbox (to the right of keep aspect ratio) is checked, when the page size values are changed, the "separator" distances between staves and systems are adjusted proportionally to the change in the vertical distance.

Note that all staff distance specifications are advisory. Dense music may result in these distances being reduced. If accuracy is required, the default points should be set accordingly.

These defaults may be changed by two sets of staff/system separation parameters (see below), which are found at the end of the Staff Lines, etc, parameter set. Most parameters apply to all relevant following music until changed (by another hidden 'PARAM' annotation). The staff separation parameters are an exception to this in that they apply to the system/staff on which they are found.

10.1. Absolute Staff/System Separation Parameters

These traditional parameters are absolute values and are independent of the layout option values (if there are different values on different voices on the same staff, the effective *average* is used):

- *"default vertical distance between one system and the next system" or "default distance between systems with separator symbols"*
- *"default vertical distance between this voice's staff and the next"*

The absolute layout parameters are useful when precise placement is needed, independent of the layout defaults.

10.2. Additional Before Staff/System Separation Parameters

These 'additional distance' parameters are in addition to the prevailing values for default separation between systems and between staves (either from the layout or as set by an absolute separation parameter):

- *"additional distance between one system and the previous system"*
- *"additional distance between this voice's staff and the previous staff"*

When determining spacing before a system or staff, Lime first applies the distance parameters found on (or before) the previous system/staff. It then looks forward for appropriate 'additional distance' parameters on the system/staff it is about to format. The *maximum* value found (if any) is *added* to the previously determined 'distance parameter'. If the additional value is positive, the default spacing will be increased, and if it is negative, the default spacing will be decreased.

If no 'additional distance' parameter is found on a system/staff, the prevailing value (either the default 0, or the value on a previous system/staff) is used. The resultant computed default distance is advisory. Dense music may result in these distances being reduced.





The additional separation parameters affect the distance to the **previous** staff/system, whereas the absolute parameter affect the distance to the next one. The additional separation parameters can be useful where cosmetic changes are wanted.

10.3. Staff Drag

The use of **Staff Drag**, whilst still supported, is hard to manage and **not recommended** except in exceptional cases. In most cases a combination of the default distances in the Layout, the absolute distance parameters, and the additional distance parameters will do everything necessary. Even if you require the drag to completely overlay another staff, it would almost definitely be better to merge staves or use 'Parts and Voices...' to put the necessary voices on the same staff, with appropriate use of uniform staff directions. Lime has long supported more than 2 voices on a staff, using internal layering.





11. Parameters

11.1. Default Values

In the parameters dialog, if a parameter value (or character value) is set as completely blank (with nothing at all in the associated edit field), the default value will be used; furthermore if there is no change before the current point in time, then the change will be deleted. The Clear button is a short-cut technique to achieve this.

If The 'Delete any changes at this time' checkbox is fully checked, then if there is an associated parameter change at the current point in time, it will be deleted and the parameter value fields are ignored.

The 'Default' button on the dialog can be used to set the associated fields to the default value. This can be useful, if one wants to change a parameter back to the default.

11.2. Normal Values

Normal values (and X-/Y-offsets for characters) are specified as decimal numbers, optionally with decimal point and (minus) sign. It is possible to specify a number as a (heavy) fraction in the form nn/dd; for example 3/2 could be entered instead of 1.5.

11.3. Character Values

The value for a character parameter is specified either as a single character or as its numeric encoding in the specified font. Encodings are identified by there being more than one character, the first one of which is a zero. The encoding can be specified either as a normal (decimal) integer, or as a hexadecimal one by preceding it with 0x (e.g. 0x20). For fonts supporting Unicode, the encoding is the UTF-16 value for the character, unless it is negative, in which case it is minus the local 8-bit coding. For music fonts (Marl, Tufa or Sonata) or the Symbol Font, the value is the 8-bit coding.

In chord symbols, the accidental font sizes are with reference to the note name being 12-points. They will be resized automatically, proportional to the note name size.

11.4. Note Head Widths

If the value for a note head width is set negative (any negative value), the prevailing value for the normal (quarter/crotchet) note head width will be used.

The new "Half Note Width" parameter actually defaults to negative and, therefore, will normally use the prevailing normal (quarter/crotchet) note head width. If you want to change this, it must be set to an appropriate *positive* value, typically close to its displayed default.

The ability to distinguish between half and quarter note head widths, allows half note heads to be emboldened as a simple technique to improve their visibility for low vision users. If half note heads are made bold, increasing the half note head width by 0.2 is recommended.

11.5. Line Widths

If a line width parameter is negative, the associated lines will not be plotted. Thus, for example, if the ledger line width is set to -1, no ledger lines will be plotted.





11.6. Parameter Notes

11.6.1. Y offset from top staff line to number of bars

This parameter specifies the position of the number of bars of a multi-measure rest from the top of the staff. The calculation uses a minimum of 5 staff lines. So if, for example, on a single line percussion stave you want to lower the number of bars, it may be appropriate to set it negative.





12. Zoom Layout

The magnification, or otherwise, of a piece may be set using the Layout dialog (in the File menu) or, temporarily, using the Zoom sub-menu of the edit menu. If the layout menu is used, the selected zoom will be saved with the file and will be used next time the file is opened. If Zoom from the edit menu is used, the change is just for the session, unless the *Edit->Zoom->Save Zoom Layout...* option is chosen

As well as predefined zoom factors such as "Small" (1), "Normal" (1.5), "Big" (2), etc, the "Frame Width" (Windows) or "Screen Width" (Mac) option may be used to specify the width of frame (Windows) or main screen (Mac). If the "Resizable" option is chosen, then the zoom size will change if the window is resized.

The Layout and the Save Zoom Layout dialogs also display the current zoom and allow the actual percentage to be set. The theoretical maximum zoom factor is 1000%, however on a Macintosh this is limited by the number of pixels needed to display a page and can be as low as 200% for large pages. If the user tries to set the zoom higher than the maximum possible, Lime will silently use the maximum.

Normal users are limited to a zoom level of 4; Lime Lighter users can use a zoom level up to 10 (6.67 times normal zoom size).

Currently the default zoom for a piece (or context therein) is set in the File->Layout dialog. This always uses the value associated with the piece options, which is not necessarily the same as the current zoom level. The *Edit->Zoom->Save Zoom Layout...* option allows the user to save the current zoom.

Unlike the full Layout dialog, the *Save Zoom Layout* dialog from the Edit->Zoom menu may be used to change and save the required zoom in a locked (published) piece.

12.1. Default Zoom

It is possible to specify the default zoom used for new scores or imported scores. Currently the facility is in the "*Preferences...*" dialog under the *Edit* menu and is fairly basic. The user can specify the zoom factor as a decimal number between 0.5 and 10, in steps of 0.5. A negative value means that the window will also be resizable.

A value greater than 10 or less than -10 means the screen/frame width. On Windows, the actual size will depend on Lime's frame size. If this is maximized, then so will any piece whose zoom is set to the frame size.

Zero (or empty) means the prevailing default, which is Normal Zoom (1.5) for normal users, and Frame Width for Lime Lighter users.

For convenience, the following characters may be used for special values:

- + Fit to screen (or Frame on Windows)
- Prevailing system default (same as 0)

Note that Lime Lighter users may specify a default zoom to be used when Lime Lighting is active (in *Edit->Preferences->Lime Lighter Preferences...*). If specified, this overrides the editing zoom for the piece.



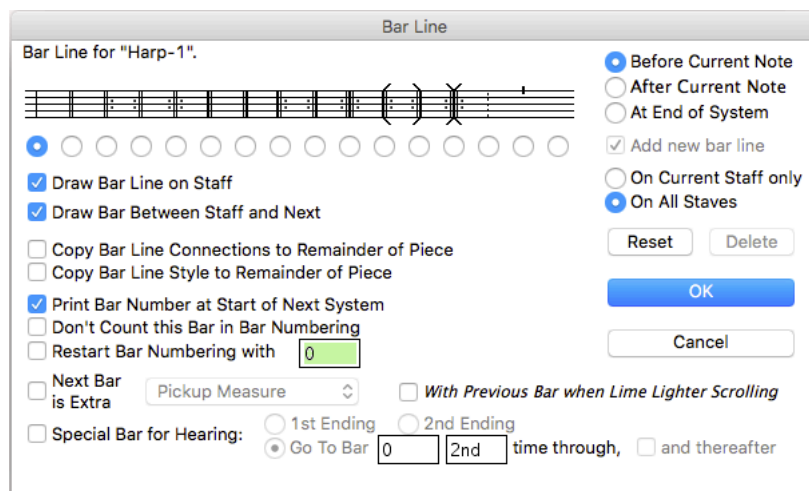


13. Bar Lines

A measure (or 'bar' in English) is a segment of time corresponding to a specific number of beats in which each beat is represented by a particular note value and the boundaries of the bar are indicated by vertical *bar lines*. Dividing music into measures provides regular reference points to pinpoint locations within a piece of music and makes the music easier to follow.

In Lime, as well as their normal musical use, bar lines provide essential control points for hearing the music and scrolling through it.

Selecting a note and using the *Symbol->Bar Line...* facility allows the user to specify bar lines at any point in the music and to change their characteristics.



Bar Line... Dialog

When the Bar dialog is first displayed, choose whether the bar line is before the selected note, after it, or at the end of system. The dialog should then display the characteristics of any existing bar line at that point.

Normally the same bar line is used across all staves, however it is possible to specify that the change is just for one staff (a common reason is to adjust whether it is drawn to the next staff, or, more rarely, if staves are asynchronous).

The following subsections summarise those fields that are not described in the old manual.

13.1. Hidden Bar Lines

If a bar is drawn neither on the staff nor between staves then it is invisible in this context. However, if you want a truly hidden bar line, the best way is to choose the empty (hidden) bar line on the far right of the options. The bar line will then be hidden and occupy no space in any context.





13.2. Hearing Options

These affect what happens when Lime is playing the piece and the bar line is encountered.

If **Special Bar for Hearing** is checked and **Go to Bar** selected, if **and thereafter** is also checked, Lime will jump to the designated Bar the specified time through if ever the bar line is encountered again. If **and thereafter** is not checked, it means that time through and only that time through. This also affects the *Take Repeat* and *Go To* of Lime Lighter.

The **Next Bar is Extra option** affects the metronome. In general Lime restarts the beat rhythm at each bar line in accordance with the time signature. An extra measure can be specified:

- **Pickup Measure.** The rhythm is just continued as per the time signature.
- **Primary or Secondary Rhythm.** Only relevant if there is an alternating time signature, this specifies an additional measure with the specified rhythm.

The **With Previous Bar when Lime Lighter Scrolling** option is only relevant if you have the Lime Lighter music stand or are preparing music for it. It specifies that when highlighting music, Lime Lighter will treat the two measures as a single entity when scrolling by measure. For example, if you have a single note pickup measure, it is often better for it to be combined the next measure rather than highlighted by itself.

13.3. Concatenated Bar Lines

Two bar lines may be concatenated together and displayed as one. In fact, if you insert a 2-way repeat bar line in the middle of a system, but subsequently split the system (e.g. in another context), the bar line will actually be split into two, with left dots at the end of one system and right dots at the beginning of the next.

This facility has been generalised so that any bar line can in fact be comprised of two concatenated lines, which are merged together. This is particularly recommended when 2-way repeat bar lines are needed.

Apart from being able to handle 2-way repeat bar lines at system breaks, the other main use of this features is to be able to have two independent 'special hear options' at the same place. For example one can have a '*Go To Bar*' 2nd time through and one 3rd time through (e.g. a repeat and then a *da capo*).

When editing a bar line that is not at the end or beginning of a system, you will have a number of options, dependent on whether the existing bar line is single or concatenated, and partially dependent on whether the line is before or after the selected note.

- **Add After Existing** If selected a new, concatenated bar line will be inserted after an existing single line.
- **Add Before Existing** If selected a new, concatenated bar line will be inserted before an existing single line.
- **Separate** If not selected, an existing concatenated pair will be edited as and subsequently replaced as a single line. The options will be a combination of both existing lines; if there is a conflict, the first one take precedence.
If selected, the first or second (depending on whether it is before or after the selected note) will be edited independently. This is useful, for example, if you want to change the hearing options for one but not the other.





14. Slurring

14.1. Slur Association

When a staff contains 'chords' in which notes are printed with different stem directions (typically because there is more than one voice) there are options to determine how slurs associate with the notes and voices.

Previous (to 9.15) versions of Lime always associated slurs to stem direction. 9.15 and later includes options for them to be more voice associated. The two main options are:

- **Stem Direction.** This corresponds to the 'normal' behaviour in Lime 9.14 and earlier. In general a slur will only terminate at a chord containing a note whose stem is in the same direction as the slur itself.
- **Voice.** A slur will only terminate on a chord containing a voice that is involved in the slur. A voice is considered involved in a slur if any chord from the beginning of the slur contained one or more notes from that voice, which had slurring set.
- **Voice and Stem Direction.** At the moment the one exception to the voice rule is that an existing slur in the stem direction will override the voice rule

The required slurring option can be set for each piece (and each context therein). The default is the original stem direction algorithm.

For MusicXML import, the combination of Voice and Stem Direction is set as this seems to give the best results in most (but not all) cases. The most obvious case where this does not work is when a voice with an outstanding slur moves to another staff; the outstanding slur then just goes on forever¹⁹! The actual algorithm set for MusicXML import is an import option.

As well as the piece/context option, each individual slur can be specified to use a specific algorithm; the default is to use the piece/context option. If one or more slurs do not seem to behave as you want them, the best way is to experiment by setting an alternative association, using the *Stem->Modify Slurring* menu..

14.2. Adding Slurs and Ties

When the slur or tie attribute is removed from or added to one or more notes, the various slur or tie related flags (such as dashed, etc) are normally reset to the default. This ensures that any vestigial flags are not applied to the new slur/tie. However, if the shift key is pressed, when a slur/tie attribute is added, then these old flags will not be reset.

14.3. Grace Slur

The Grace Slur sub-menu item specifies whether to draw a slur appropriate for grace notes. This is set by default for grace notes. For ordinary notes this can be useful when slurring from a regular note to a *nachschlag*.

If one of the stem menu items to add a slur or additional slur to a group of notes (i.e. more than one selected), if one or more of the notes are flagged as grace slurs, then the flag will be set for all notes in the group.

¹⁹ Clearly this needs to be detected and resolved!





14.4. Slur End Placement on Stem or Note Head

Lime will place the beginning and end of a slur either near the stem of the associated chords or close to the note-head. By default Lime will choose the position that looks best considering stem directions, beaming, etc. Two menu options under *Stem->Modify Slurring* allow some control over how slurs that begin and/or end at a stem are placed with respect to a particular chord:

- Near Note Head* Specifies that they should actually be drawn as near to the note head as possible.
- Near Stem End* Specifies that no attempt should be made to move the end closer to the note head.

[As of 9.17, it will be possible to specify these on the terminating chord as well].

14.5. Ties as Slurs

The normal use of the *Tie* attribute on a note is to specify that the note is tied to the next note in the same position in the next chord; when played the two notes (or more if the tie is extended) are sounded as one continuous long note.

The tie attribute can also be used as an indication of a split slur²⁰. If there is no equivalent note-head in the next chord, Lime will draw the tie as a sort of split slur (a 'pseudo-slur') to the nearest note-head *in the same voice*, if there is one. For any particular note-head, only one 'pseudo slur' will be drawn terminating on it.

14.6. Conversion of Old or New Files

When old files (pre 9.15) are upgraded, Lime attempts to ensure that, as far as possible, slurring looks the same after the upgrade. However, if there are voices that print on the same staff in one context, but on different staves in another, it is possible for there to be a slight change in where slurs are terminated in the context where they print together, if stem directions have been specified explicitly. In general, the changes are for the better; but they are changes.

Similarly, when new files (post 9.16) are downgraded, there may be minor differences in the exact position of slurs.

²⁰ The primary purpose of such tied slurs is to support MusicXML import.





15. Hidden Objects

15.1. Hidden Annotations

When a hidden text annotation is displayed or printed, everything behind it is masked out, so the annotation can be clearly seen.

Lines and Curves may be hidden in the same way as Text can be. If a hidden line or curve is displayed or printed, it will be in light grey. Hidden lines and curves are probably only useful as part of fixing old scores.

15.2. Hidden Symbols

Bar Lines, Time signatures, Clefs and keys can be hidden (using their dialogs). Hidden symbols and rests can be seen temporarily using the *Symbol->Show Hidden Symbols* menu option. This will show them in light grey. Note that there will be changes in the music layout to accommodate the hidden symbols, while they are being shown. Showing hidden symbols allows you to see where they are and modify them. This is particularly useful for showing hidden rests, and for showing hidden Bar Lines, Key Signatures and Time Signatures when they can significantly impact play-back and lay-out.

15.3. Hidden Rests

Showing hidden symbols also shows hidden rests, but only those that are explicitly hidden and actually occupy space. If there is more than one voice on a staff, one of which has hidden rests, more likely than not these rests will be discarded and not occupy any space. The *showing hidden symbols* option will not show them.

If a rest is explicitly hidden, Tuplet marks will never be shown on it, whether or not the *Hide Tuplet* flag is set for it. However, unless the *Hide Tuplet* flag is set, a hidden rest will be considered when tuplet groups are identified. If the *Hide Tuplet* flag is set for a hidden rest, then the rest will be totally ignored for tuplet purposes.

Note that, if there are multiple voices on a staff with rests at the same place, if they should not be seen, they all need to be hidden, otherwise one will be shown.

15.4. Hidden Clefs

A Clef can be hidden, but this only applies at the point of insertion; the duplicates, put at the beginning of the following system will be shown as normal.

In the *Symbol->Clef...* dialog, if the Hide Clef checkbox is partially checked, the clef will be hidden unless it is at the beginning of a system. If it is fully checked, it is always hidden (not recommended, except in special cases).

If a Clef on a staff would duplicate an earlier one in the same measure on the same staff, it will not be plotted (it will be hidden automatically). This can happen in rare cases if two or more asynchronous voices are printing on the same staff.

15.5. Hidden Key Signatures

A Key Signature can be hidden, but this only applies at the point of insertion; the duplicates, put at the beginning of the following system will be shown as normal.





In the *Symbol->Key Signature...* dialog, if the Hide Key Signature checkbox is partially checked, the signature will be hidden unless it is at the beginning of a system. If it is fully checked, it is always hidden.

15.5.1. Default Accidentals

A hidden Key Signature is useful, if there is a need to set permanent default accidentals, where the actual key modulates but the signature remains the same. Place a hidden key signature wherever you want the default accidentals to change (see section 21.4.2).

15.5.2. 8^{va} and 8^{vb}

A hidden Key Signature is also particularly useful for temporarily setting octave-shifted play-back, such as 8^{va} and 8^{vb} . Place a hidden key signature before and after the required section and specify the required octave shift (see section 21.5)

- *before* specify +1 for 8^{va} , -1 for 8^{vb} , etc (up to ± 3). This adds a temporary octave shift over and above any existing transposition;
- *after* revert the transposing to what it was before the section.

15.5.3. Showing Invisible Key Signatures

A key signature with no accidentals (C-major or A-minor) or natural changes is normally invisible. When *Showing Hidden Symbols*, such invisible keys are shown as two naturals one above the other, as shown below.



Symbol for Showing C-Major or A-Minor Key Signature





16. Preferences, Context Options and Piece Options

Preferences and Options are now three separate dialogs.

- Preferences.* Currently under the Edit menu, the general preferences are those for the use of the Lime program that are piece independent.
- There are additional preferences dialogs for Lime Lighter and Lime Aloud.
- Options for Context.* Currently under the file menu, the context options are for one or more notation contexts in the current piece that is being edited. The context options are handled in the same way as the layout attributes.
- Options for Piece.* Currently under the file menu, the piece options are for the current piece that is being edited. They affect every notation context.

16.1. New Preferences

16.1.1. Make ~Backups when Saving

If checked, whenever a file is saved, the previous version is retained with a tilde (~) before the first character. If it is fully checked, this also applies to 'save as'.

16.1.2. No Warnings when Opening Older or Newer File Formats

By default if a piece is opened that has an older or newer file format, the user is warned when the file is opened. If this preference is fully checked, then there will be no warnings; if it is partially checked, there will only be warnings for newer file formats.

Note that, if a piece was in an older or newer file format, 'save' will always act as a 'save as'.

16.1.3. Emboldening Preference

The *Emboldened* preference can be used to generally embolden the score when displayed and, optionally, when printed. When turned on, this is equivalent to setting the minimum line size for many elements to 0.8 and using parameters to embolden things like half note heads (minims) and dots, but without actually modifying the score. Emboldening works best when the zoom level on the screen is *Big* (2) or larger.

When partially checked, this preference affects the display only; when fully checked printing will be affected as well.

Lime Lighter users can control the individual features - see Lime Lighter preferences (section 24.8).

16.1.4. Metronome is default when Hearing

If this is checked a metronome will be the default for the session when the Hear option is used; if it is fully checked, then a lead-in will also be default.



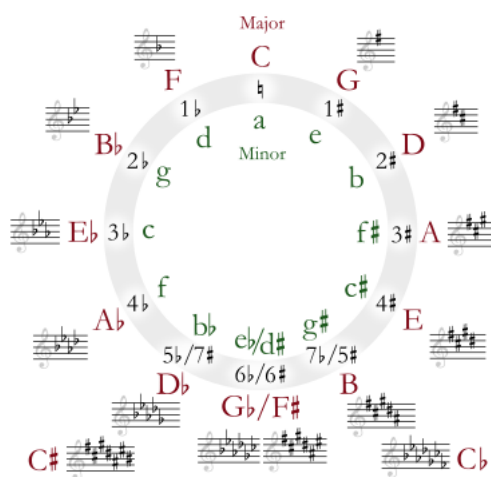


16.1.5. Highlight Notes with Unplayable Pitch

As of 9.16, it is possible to specify the playable range of one or more voices or parts, using the *Voice->Voice on Channel* dialog. If the *Highlight Notes with Unplayable Pitch* preference²¹ is active, any notes outside the specified range for the voice will be highlighted (in colour number 2 in the preferences colour palette, usually in Red - see section 9.2).

16.1.6. Old Default Accidentals

Prior to 9.16, the default accidentals were based on the number of sharps or flats in the prevailing key signature and treated the key as a major (even if it were a minor key). Furthermore, the accidentals for the sharp keys (and C major) were shifted so that, for example in C major, an A flat was preferred to a G sharp). In 9.16.3 and later, the default accidentals are chosen so that accidentals from neighbouring keys in the circle of fifths are preferred and, moreover, a distinction is made between major and minor keys.



Circle of fifths showing major and minor keys (from WIKIPEDIA)

The *Old Default Accidentals* preference is for diehards who are used to the old (incorrect) Lime default accidentals. If it is set, the old, pre-9.16 default accidentals will be used by default, if default accidentals have not been set for the prevailing key signature. Note, however, that in the *Default Accidentals...* dialog, the Defaults button will always set the *new* default accidentals, irrespective of the *Old Default Accidentals* setting.

As of 9.16.7, Default Accidentals can be permanently associated with a key signature.

16.1.7. Music Window Flashing Rate (of insertion point)

This is normally set to the flash rate per second. If it is set to 0, the insertion point will not actually flash at all, but just be shown in the flashed state. This may occasionally be useful if a screen reader does something funny with flashing text on the screen.

²¹ The preference to 'Identify Note with Modified Pitch' highlights (usually in Orange) whose pitch has been changed with *Hear->Higher* or *Lower*, is still available. However, the old facility to identify notes with modified volume has been deprecated.

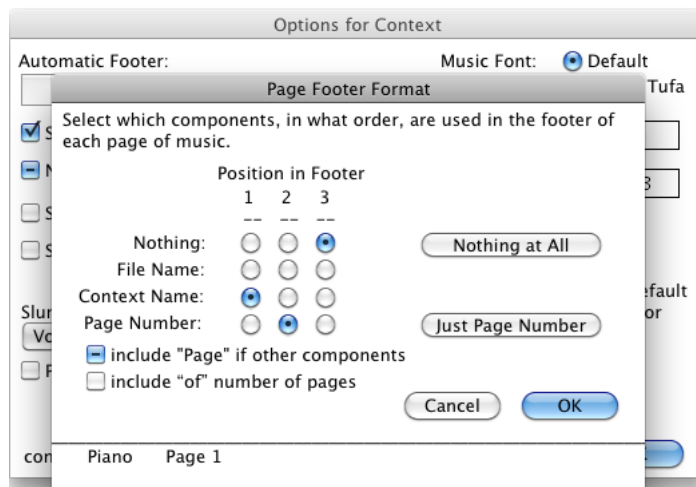




16.2. New Context Options

16.2.1. Automated Footer

The two context options to 'Show Page Numbers' and 'Show File Name on Each Page' have been replaced by an *Automated Footer* option, which allows for a combination of Page Number, notation context name and file name (without any .lim extension). This is handled by a footer sub-dialog as shown below. The radio buttons can be used to specify what components are shown and, as a convenience there is a button to set just the page number and there is one to set nothing at all.



Setting Automated Footer

If page numbers are shown, the 'include "Page"' checkbox can be used to specify whether the word "Page" is included:

Unchecked: never include "Page";

Fully checked: always include "Page";

Partially checked: only include "Page" if there are other components.

If checked the "include "of" number of pages" will include "of" the number of pages in the context.

An example of the selected automated footer is shown at the bottom of the dialog²².

16.3. New Piece Options

16.3.1. Uniform Marks as in 9.05 and earlier

Lime normally places marks such as staccato, tenuto or accents for a note, near the note head if possible. Where there are notes with different stem directions, marks will be placed near the stem instead. In Lime 9.05 and earlier, at any point where there were notes with different stem directions on the same staff, marks and accents would always be plotted on the stems to avoid any clash. After 9.05, this does not happen if the only notes in the other direction are invisible (e.g. invisible rests).

The *Uniform Marks as in 9.05 and earlier* option, specifies that, in such cases, the marks will always be placed on the stems.

²² n.b. the file name shown in the dialog does not yet support exotic file names (Unicode)





17. Polymetric and Polytemporal Parts

Lime 9.15.9 includes support for system separation and pagination in pieces which have parts that proceed with multiple independent beat rates and whose measures do not align because they are of different lengths. This can legitimately happen for two reasons (within Lime the term '*Polymetric*' often means one or the other or both):

- The parts have incompatible time signatures such that the effective length of each measure is different. The playback length of a note (e.g. a quarter/crotchet) is the same in each part, but there are a different number per measure (for example, one in 4/4 and another in 3/4). This is called '*Polymeter*'.
- A '*Polytempo*' note duration multiplier (see section 17.4) is used in one or more of the time signatures. This modifies the effective length of each notated note. For example if the multiplier on part A were 1.5 against 1.0 in part B, a quarter/crotchet in part A would last the equivalent of a dotted one in Part B. In the score the different lengths will be reflected in the spacing.

If the differences between the parts in a context (e.g. the score) are significant, it can be impossible to put system/page breaks at measure boundaries, which align on all parts. Prior to 9.15.9, Lime would just go ahead and split at an available measure boundary. This meant that after the split, the parts were not aligned by time, rendering them impossible to read/follow together. Lime 9.15.9 now allows a staff to be split anywhere, even in the middle of a chord. After the split, everything remains aligned.

There is an option for a piece's context (options for Piece... menu item) that specifies that the piece has *Polymetric Parts* (or polytemporal) and system breaks can be inserted anywhere, even if not at a timing boundary.

The following extract shows a very simple (silly), polymeter example of 4/4 against 3/4. There are 3 measures on the first staff and 4 on the second. There is no measure on either staff, where the system can be split at a measure boundary on both staves. Furthermore, a split at the 3rd measure in the 2nd staff is not at a timing boundary on the 1st staff.

Staff 1 is 4/4

Staff 2 is 3/4

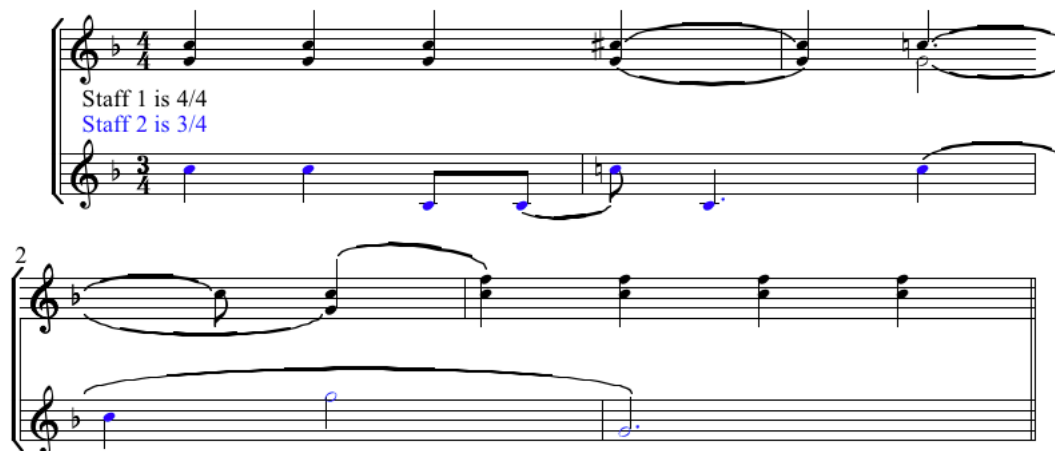
want to break 2nd staff here.

Lime can, however, split at this point and correctly align the music afterwards, as can be seen below. What happens is that any note that starts before the time of the system break, will be placed before the break. The notes on that staff after the break will be placed according to the amount of time remaining and, therefore, will be correctly aligned with the other staff, as can be seen in the example below.





When a system break is not '*Bar Associated*' at a measure boundary, there will not be a bar line at the end of its staff before the break; furthermore, there will not be a line at the beginning of the staff after the break (though this can be controlled by a parameter).



Staff 1 is 4/4
Staff 2 is 3/4

Result of Splitting a System with Polymetric Parts

17.1. Polymetric Parts Option

In most music, parts should align at measure boundaries (or at least at regular timing boundaries) and, therefore, users are warned if there appears to be a problem. Setting the *Polymetric Parts* option (*File->Options for Piece...* menu item) avoids these warnings, and tells Lime that the context has Polymetric or Polytemporal (or both) parts.

The *Polymetric Parts* option is context specific. In most cases, contexts for individual parts will want systems split at measure boundaries; it would probably not be appropriate to set the *Polymetric Parts* option for those contexts.

Whether or not this option is set, there is no restriction on where system breaks may be placed (other than avoiding a completely empty system). Its main use is to warn normal users of possible mistakes and, furthermore, to set the default in some of the dialogs to be '*current staff only*' rather than '*all staves*'.

17.2. Note Editing

When editing notes at the end of a system (including end of page), which is not *bar associated* (does not occur at a measure boundary), if note lengths change, Lime will automatically adjust which notes go before the system break and which after it.

17.3. Beaming

Currently, if you want a beam to extend over a *non-bar associated* system break, you will have to set the notes before the break to *Continue Primary Beam* explicitly (using the *Note* menu).





17.4. Polytempo Note Duration Multiplier (very advanced feature)

If a time signature has a *Polytempo Duration Multiplier*²³, the duration and spacing of every note after this time signature is multiplied by this value. In essence, the multiplier is applied to the effective note length. If, for example, the multiplier is 1.5, then a quarter note (crotchet) will take the same time as a dotted quarter note on a part without a multiplier. Parts with different multipliers run at different paces through time.

If a polytempo duration multiplier is needed, it can be specified as part of the time signature, using the advanced '*Polytemporal*' attribute. If you do not actually want a visible time signature, then it can be hidden (placement, hidden) and, moreover, a signature of 0/0 is allowed, when there is a polytempo multiplier.

In most cases, there is a well-defined ratio between polytemporal parts. It is, therefore, possible to specify and maintain the *Polytemporal Multiplier* as a ratio in the form of a (heavy) fraction, or even as the product of two fractions. If something more irrational is needed, then it can be specified as a positive decimal number.

Polytempo Duration Multiplier specified in a Time Signature

If the fractional form is used to specify the multiplier and the fractional radio button remains selected when OK is pressed, the multiplier is maintained and displayed in the fractional form. If, however, the decimal button is selected, the multiplier is maintained and displayed as a simple decimal.

If an existing multiplier is in decimal form and the fractional option is chosen, Lime will attempt to identify a fraction that is within less than 0.00001 of the original.

²³ The Polytempo Duration Multiplier is a very advanced feature aimed at supporting certain types of modern music (www.theoctetproject.info was an example but is, sadly, no more). You will not normally want to use it and, unless you know what you are doing, it is probably a mistake to set it at all.





17.4.1. Re-computing Playback

The *Polytempo Duration Multiplier* affects the actual playback time of each chord. If the multiplier is changed for any reason (change in value or deletion of a time signature) then it is necessary to re-compute playback durations for the affected notes. This is done automatically.

17.4.2. Linear Spacing

In a truly polytemporal score, using different *Polytempo Duration Multipliers* in different parts, linear spacing of the score is not recommended. There is no problem, however, in parts' contexts, providing the multiplier is the same.

In 'normal', measure aligned music, it is straightforward to align multiple staves and to space the music linearly in time. However, if anything other than very simple multipliers are used, it is seriously difficult to align music linearly within the constraints of a page because of the need to provide space for the bar lines on each staff. Unless the page width is big enough, more often than not, the music will be over compressed in order to fit.

17.4.3. Upgrading Old Scores

Prior to Lime 9.15.9, the *Polytempo Duration Multiplier* was actually an obscure parameter. When reading such scores, Lime 9.15.9 and later, automatically sets the *polymetric* option and converts the parameter to the time signature attribute. If there is no time signature near the parameter annotation, then a new, hidden one will be created.

When Lime 9.15.9 saves a score with polytemporal parts, the old *Polytempo Duration Multiplier* parameters will be restored, so that the score is fully compatible with earlier versions of Lime.

17.5. Systems...

System/page breaks can be inserted with the *Page >Systems...* facility or manually at the desired using *Page >Split System*. To get alignment, you must set up the system breaks AFTER you have set up all the staves and, more important, AFTER you have entered the multiplier. The initial creation and insertion of measures does not do it automatically. Moreover, note entry does not change the position in time of any system breaks; if the break is at a bar line then that is where it stays, aligned or otherwise.

The easiest way of setting up the system breaks is to select the beginning of the fastest stave (i.e. the one that has most measures) and use *Page->Systems*, and select the '*New Systems and Page Start Locations (Fixed)*' option.

17.5.1. Bulk Breaks with the Page->Systems... option

The Page->Systems... option always works on the Voice of the note that is selected when the dialog is invoked. Measure breaks, etc, are all with reference to that voice. The systems that are created will all be at a measure boundary in that voice.

In 'normal' pieces where all parts have the same measures, it doesn't really matter which voice is selected. However, with *Polymetric Parts* (and polytemporal) it is important that the correct voice is used. It is recommended that the voice with the most measures is used as the master.

It is important to note that in order to get alignment you must set up the system breaks AFTER you have set up all the staves and, more important, AFTER you have entered the multiplier. The initial creation and insertion of measures does not do it automatically.





Currently, use of the Systems... option for variable system and page starts based on note density in contexts using different polytempo multipliers is not recommended as it can lead to strange results. However there are no problems with any of the fixed options.

17.5.2. Manual Insertion of System and Page Breaks

System/Page breaks can be inserted manually anywhere in a piece using the Page->Split System menu option (except, of course, at the start of a system). The breaks will be inserted before the selected chord(s), whether or not it is at a measure boundary.

By default the break will be inserted at the same time on all other voices. If this has not been designated a *Polymetric Parts* context, the user will be warned if the break is not at a timing boundary on one of the other voices with the option to cancel the attempt. This warning includes the option to convert the context to being a *Polymetric Parts* one.

If you are a die-hard and really want to have system breaks at measure boundaries and nowhere else, even if the measures do not align, it is still possible to do so by invoking the Page->Split System menu option with the Shift key down. In this case the break will be inserted at the beginning of the current measure on the selected voice and on other voices at the same place, or the next measure after if they do not align. This is not recommended; it is only for backwards compatibility and may be rescinded in 9.16.

Manual staff splitting can be done at more than one place at the same time, by selecting more than one note. Care does need to be taken that the selected places do not conflict. If the results are not as expected, undo and try again.

17.5.3. Measure Insertion and Deletion

In *polytemporal* pieces, a fixed number of measures should only be inserted or deleted on all staves in exceptional cases. Normally, it is necessary to do it staff by staff. If measures are inserted or deleted, it will probably be necessary to redo system breaks.

17.5.4. Further Study

The polymetric and polytemporal parts capabilities are 9.16 facilities. Any piece, taking advantage of the facilities will not be rendered properly in 9.15.8 or earlier.

The following aspects still need to be addressed:

- Automated handling of beaming over system breaks that are not at measure boundaries.
- Time signature for more than one but not all staves.
- Remembering the master voice used for creating systems and, perhaps, giving the user the chance to use it instead of the selected voice for the *Systems...* option.
- There are issues with linear spacing in pieces using polytempo multipliers with many staves (bar lines can get overlapped). Normal spacing is, therefore, recommended.
- Use of the *Systems...* option for variable system and page starts based on note density in contexts using different polytempo multipliers is not recommended. However all fixed options work nicely.
- Can anything sensible can be done to insert or delete compatible measures on all staves? Currently in polytemporal pieces, the only practical way is staff by staff.

Note that before 9.15.9, the Polytempo Duration Multiplier was an obscure parameter in the staff section; after 9.15.9, it is an attribute of a time signature and properly integrated.





18. Parts and Voices...

18.1. Parts and Voices... Dialog

18.1.1. Arrangement Propagation

One of the options in the (complex) '*Parts and Voices*' dialog is whether to propagate the arrangement beyond the current point in the piece. Such propagation overrides any changes until the designated position. The options are:

- Until the End of Piece* All subsequent arrangement changes are removed (prior to 9.16.3, this was the only propagation option).
- Until End of System* The Change will be propagated to the end of the current system and then, unless there is another change there, reverted to what it was before.
- Until Bar Number* The change will be propagated to the specified *printed* bar number and then, if there is not another change already there, reverted to what it was before.

The bar number is the *printed* bar number. If the specified bar number does not exist after the current position, then propagation will be to the end of the piece.
- Selection Only* If the *Selection Only* checkbox is checked, the propagation will only be done for the currently selected staff. This can be useful if other staves have voices that change staff.

18.1.2. Apply

The '*Apply*' button applies outstanding changes but stays within the dialog, allowing further changes and use of the *previous* and *next* buttons.

18.1.3. Previous and Next

The '*Previous*' and '*Next*' *arrangement change* buttons go to the previous and next changes in voice arrangement in the current context (on different page if necessary), cancelling any unapplied changes.

18.1.4. Don't Print

In the *Parts and Voices...* dialog, if none of the propagations options is set, when a staff is set as *Don't Print*, there is an option to set it for the rest of the piece for the voices on the staff only. If the *Don't Print* check-box is fully checked, it will not print until the end of the piece; if it is only partially checked (— on Macs or a dimmed tick on Windows), then the effect may be rescinded later in the piece. In the staff list, a stave that doesn't print until the end of the piece is indicated by an equals sign, but if it may be rescinded later it is indicated by a minus sign.

18.1.5. Move Staff...

The *Move Staff* sub-dialog allows one to change the staff number of the selection. Unless the *Number in List* checkbox is checked, the number is the actual staff number required. If the selection is not-printing (staff 0) the default is to use its position in the list, the top being position 1. If staff is used and the selection is printing, any following non-printing lines, with voices in the same part, will be moved as well.





18.1.6. All Staves

The '*for All Staves*' check-box in the *Parts and Voices...* dialog specifies that the staff size should be applied to all staves (when OK is pressed). If it is only partially checked (— on Macs or a dimmed tick on Windows) then, if a voice changes the specification elsewhere in the piece, the user will be asked whether the change should be done for all of this voice and all remaining voices. If it is fully checked, the change will be done for all voices for the whole piece.

18.1.7. No Ledger Lines

The '*No Ledger Lines*' check-box in the *Parts and Voices...* that no ledger lines should be plotted for notes on the designated staff. Normally this will only ever be used if there is only one (or no) staff lines, so the user will be warned if this is checked and there is more than one staff line. Note that this is a 9.16 feature and will have no effect if opened with Lime earlier than 9.15.9.

18.1.8. Changes to Number of Staves

For obvious reasons, the number of staves in a system should not change in the middle of a system. If they do, Lime will actually print the maximum number of staves but inconsistent staves will have empty portions.

The *Parts and Voices...* dialog can detect and warn about an inconsistency in the number of staves in a system at the beginning of a change and most cases of one at the end of a change. If a potential inconsistency is detected, the user will be warned and offered the chance to cancel the change. The change may, of course, be correcting an existing inconsistency; in which case, it would be reasonable to allow it. However, normally, it will indicate a mistake.

18.1.9. Part or Voice Change

If a Part or Voice is added or deleted, it takes effect immediately and cannot be cancelled. Furthermore, it cannot be undone and, moreover, undoing is reset!

Part manipulation (adding, deleting or renaming) can only be done within the main, score context. In other contexts, the associated buttons are not available and replaced by the context name.

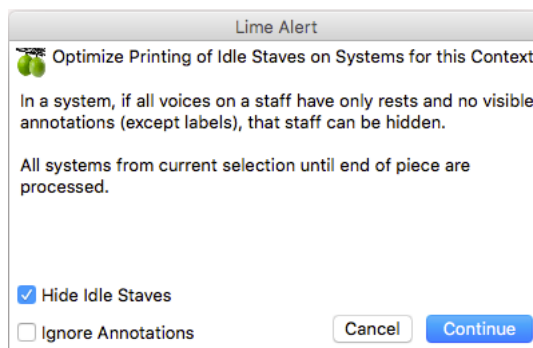
18.2. Idle Staff Hiding

Many scores, particularly for orchestras, often contain parts that are silent for a significant number of systems, often for several pages. It is often desirable to omit these parts from the score when they are not playing. Lime provides a simple, automated facility to identify parts that are idle on a system and not to print their associated staves in the current context. This is invoked using the "*Voice->Hide Idle Staves*" menu option. The user is then given the option to proceed and hide or show empty staves in all systems beginning with the one containing the current selection.

A staff is considered "idle" if all voices that use it are "idle" for the whole system. A voice is considered "idle" if it has only rests and has no important, visible annotations. The staves used and the parts (voices) they contain are those that have been defined using "*Parts and Voices...*". A part is "idle" if all its voices are "idle". On a system by system basis a staff will be hidden if all the parts that use it are "idle".

By default all visible (not hidden) annotations are considered important except any duplicated on each system and any classified as a label (e.g. part name). However, when hiding idle staves, there is an option to ignore all annotations.





Idle Staff Hiding Invocation

Currently²⁴ the only other option available (apart from ignoring annotations) is whether idle staves will be hidden or shown. If the *Hide Idle Staves* check-box is set then all idle staves will be hidden. If it is cleared (un-checked), then all staves will be shown on all systems, including any that are not printing (hidden). Processing starts at the beginning of the system containing the current selection and continues to the end of the piece.

When processing is complete the user is told whether any changes were made and, if changes were made, an indication of how many idle staves were identified, whether or not their status was changed; a staff is counted in each system analysed. If any changes were made, the user is given the option to re-paginate the context from the current page automatically.

It is strongly recommended that the results are checked on every affected page, particularly if parts start and stop or hop between staves as there are known issues with such parts.

Notes:

- Currently, if a part uses more than one staff, then either all staves are hidden or all are shown, when or not some are idle. For example, if only one of the staves on a grand staff part is "idle", both staves will still be shown. If you want to allow one of a part's staves to be hidden, then the part would need to be split.
- When hiding staves, in the unlikely situation that all staves on a system are "idle" and would be hidden, then one of the empty staves will be shown, if possible the staff containing the original selected voice, otherwise the last one.
- The idle staff hiding algorithm works on the staff parts and voices at the beginning of each system. If there is a change mid-system whereby a part hops to another staff or another part hops to a hidden staff, the resulting staff is unlikely to be desirable.
- If a staff on a system is hidden, any staff dragging will be removed and will not be restored if the staff is shown again. Staff dragging is, in any case, not recommended, see section 10.3.
- If changes were made, it will probably be desirable to repaginate so as to make optimum use of available space on each page.

²⁴ If required, in future versions, it would be possible to specify a subset of parts to process and, furthermore, whether parts can be partly shown. If required, it would also be possible to specify a stop point (bar) for the processing.





19. Quarter Tones and Special Accidentals

The *Note->Special Accidentals...* menu item can be used to show (or hide) special accidentals, such as quarter tones, on the piano window (short-cut is *ctrl/cmd->single quote*). Special accidentals can be used either in note entry mode or to modify the accidentals of an existing note.

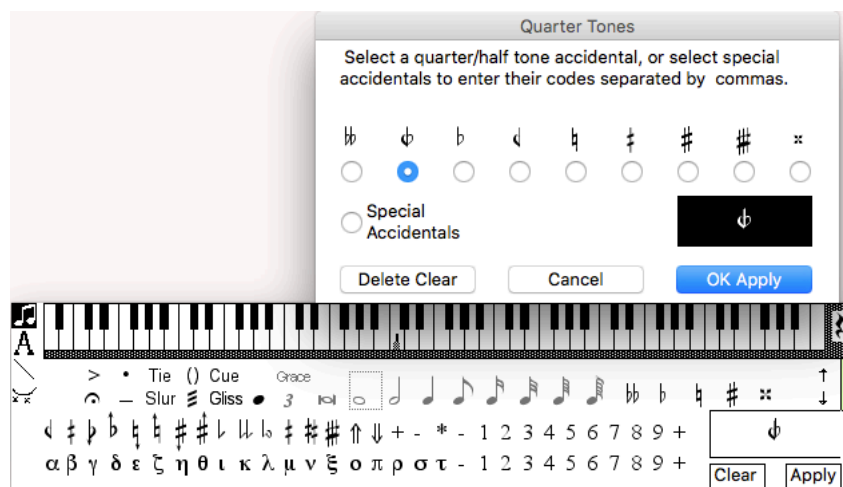
- Note entry mode Select the required accidental(s), then play the associated note.
- Modification mode Select required note(s) then select the required accidental(s), then click the *Apply* button.

The actual symbols and associated pitch modifications can be changed using parameters. It is, however, very strongly recommended that the first 16 special accidentals are not changed as these are commonly used quarter tone accidentals.

When the piano window is showing special accidentals, if one or more special accidentals are selected, they are shown in the box above the *Clear* and *Apply* buttons. Select special accidentals by clicking on the required ones in the extended piano window. Standard semi-tone accidentals can be included by right clicking one. If a standard accidental is left (normally) clicked and no special accidentals are currently selected, it will be applied as normal²⁵.

19.1. Quarter Tone Special Accidentals Dialog

Whether or not any special accidentals are selected, the display box above the *Clear* and *Apply* buttons, which shows any selected special accidentals, is itself a button, which if clicked, will bring up an accessible dialog, designed to make it easy to apply quarter-tone accidentals. When the piano window is showing special accidentals, *Shift-* (shift-backslash) will invoke this special dialog, provided the selected note is not a rest or when Lime is in note entry mode.



Accessible Quarter Tone Dialog and Piano Window with Special Accidentals










In note entry mode, right clicking the display box will simply select the same special accidentals that were used when the last note was entered (if any). The same effect can be achieved by hitting the short cut (*Shift-*) very fast.

²⁵ The need to use right click to select a standard accidental as the first special accidental is a slight change in functionality, first introduced in 9.16.9.





The radio buttons allow one of the quarter/semi tones to be selected. They are from left to right:

-  Double (whole tone) flat,
-  Three quarter flat,
-  Ordinary (semitone) flat,
-  Quarter flat,
-  Natural,
-  Quarter sharp,
-  Ordinary (semitone) sharp,
-  Three quarter sharp,
-  Double (whole tone) sharp.

If the *Special Accidentals* radio button is selected, an edit box appears in which one or more special accidental numbers may be selected as if they had been clicked in the piano window. In the unlikely event that they are required, the 'standard' accidentals may be specified using + and - as follows:

- + Ordinary (semitone) sharp,
- Ordinary (semitone) flat,
- +- Quarter flat,
- + - Quarter sharp,
- ++ Double (whole tone) sharp,
- Double (whole tone) flat,
- ++- Three quarter sharp,
- + Three quarter flat,
- 0 Natural.

If required, more than one special accidental can be selected; all selected ones are displayed in the box above the *Ok Apply* button in the quarter tone dialog and above the *Clear & Apply* buttons in the special accidentals piano window.

When a special accidental is applied to a note, the pitch of the note is changed as per the accidental.

The standard accidental buttons (and, therefore, short cuts) have different effects depending on whether the piano window is in "*normal*" mode or "*special accidental*" mode:

"normal" Clicking an accidental (or using the short cuts), adjusts the notation if possible, while retaining the pitch. For example, if the selected note is G-sharp and the standard *Flat* is clicked, the notation will change to A-flat.

"special accidental" Left-clicking a standard accidental (or using the short cuts) has the same effect as in normal mode, provided no special accidentals have been selected. However, right-clicking a standard accidental simply adds it to the selected specials. Left clicking one will only add it if there are already selected special accidentals. Any accidental entered using the quarter tone dialog is deemed to be *special*.

In "*special accidental*" mode, clicking a special accidental, or right clicking a standard one adds it to the selected accidental set. When these selected special accidentals are applied (there is an *Apply* button), the pitch of the notated note is changed to match the sum of the accidentals, while the notated note (i.e. position on the staff) stays the same.

If more than one special accidental is specified, when they are applied, the note's pitch is adjusted by their sum. In fact, the three quarter flat, used in the quarter tone dialog, is a combination of a quarter flat and a flat.





19.2. Note Entry

In note entry mode, the selected set of special accidentals (if any) are applied to the next note that is entered. After a note has been entered, the special accidentals (and quarter tones) are cleared and the standard “*Natural*” is selected. Unless another accidental is specified, the next note to be entered will be as normal. If, however, the natural is cleared (clicking on the standard natural box in the piano window or using the short cut, back-slash) and a note is entered, if there is a previous note in the same measure with the same base notation with a special accidental, it will be applied to the new note with associated pitch adjustments²⁶.

19.3. Hearing Special Accidentals

Unfortunately, the MIDI standard (even in 2020) does not support anything other than semi tones, except by using pitch bend. Thus, in order to hear microtones when playing the music, the voice must be set appropriately. This is done using the *Voice->Voice on Channel* dialog, specifying *Microtonal Tuning* and, usually, *Using Pitch-bend* with the number of channels set to the maximum number of notes that are ever played simultaneously on that set of channels.

If quarter tones or other special accidentals are selected when a piano key is clicked, the appropriate pitch will be sounded, whether or not the voice is normally set for microtonal tuning. If it is not set for microtonal tuning, it will be temporarily set with just one channel, which means that only one note can be sounded at once and, thus, only the last note in a chord will be heard.

19.4. Special Accidentals Parameters

The quarter tone dialog relies on the pitches of the quarter tone accidentals not having been changed, particularly for quarter flat, quarter sharp and three-quarter sharp (special numbers 1, 2, 14). It is strongly recommended therefore that if practical, the ‘cents offsets’ of the first 14 special accidentals are not changed.

However it is recognized that, for some uses, it is necessary to change the definitions and offsets of most of the special accidentals from 10 and upwards. In that case, it is recommended that special accidental 9 (♭) is changed to the ‘standard’ three-quarter sharp (♯, with 150 cents offset)²⁷. Lime will detect this and use it instead.

²⁶ This used to always happen regardless of the standard accidental status; since 9.16.9, it will only happen if no standard accidental (including Natural) is selected. It is retained for partial backwards compatibility.

²⁷ The ‘standard’ 3/4 sharp is a lower case ‘n’ in Tufa font, 24pt and with a y-offset of -2.





20. Ticker-Tape (panorama) and Autocue Contexts

*Ticker-Tape*²⁸ and *Autocue* are viewing modes designed to facilitate continuous scrolling measure by measure or system by system. A ticker-tape notation context can be scrolled measure by measure like a 'ticker-tape', with no visible system or page breaks, and an Autocue one can be scrolled vertically system by system. The purpose is to enable the user to always see the next measure(s) even if they would be on another system or page in a normal page-based score.

Ticker-tape is often referred to as "panorama" view. It allows the music to be seen across system and page breaks, which often makes it easier to analyse and appreciate its structure. It can also be useful in a duration Fixer context as it allows bar misalignments to be seen across system and page breaks (see section 8.1).

It is, of course, possible to set up a context within a short piece, which comprises a single, wide page with one wide system. However for long pieces, memory constraints (even with today's massive computers) can cause problems and performance limitations. Ticker-tape overcomes these limitations by restricting the actual pseudo page (known as the ticker-tape portal) to the width of just 3 normal systems.

Ticker-Tape and *Autocue* are set up as separate contexts, whose systems are defined for it. In ticker-tape and autocue contexts, the music is divided as normal into systems, but there is just one system per nominal page. If the context were displayed in normal mode, there would be usually be just one system per page.

They are very easy to set up using the *Duplicate Context* button in the part extraction (*Context->Part Extraction...*) and specifying the required context type.

In a horizontal *Ticker-Tape* context, the music is displayed as a semi-continuous single system. Each, single "system page" is actually displayed concatenated with the system (page) before it and the system (page) after it, as though the three systems were one long one (and just one system on the page). The nominal page number is the middle system, except at the beginning and end of the piece. The first system page in the piece is actually concatenated with the next two; similarly the last system page is concatenated with the previous two. In both cases, this is, of course, only if there are three or more systems in the parent context.

In a vertical *Autocue* context, the music is displayed as semi-continuous pages, with three systems per page portal. When tracking in Lime Lighter, it will always track by system, regardless of preferences.

20.1. Setting Up a Ticker-Tape or Autocue Context

A convenient way of setting up a ticker-tape or autocue context is to use the '*Duplicate Context*' feature of the Part Extraction dialog, setting the appropriate type of context in the sub-dialog. This creates the new context of the required type, containing the same parts, system breaks, etc as the original, except that staff drags (which are, in any case not recommended) are not duplicated if the type of context (Autocue/Ticker-tape) is changed.

²⁸ Ticker-tape and Autocue context were originally designed for the Lime Lighter music stand.





Setting (or clearing) a ticker-tape or autocue context may also be done using the *Systems...* dialog, from the *Page* menu. You simply set or clear the appropriate context checkbox. Note that the score context cannot be set as ticker-tape or autocue (the check-boxes will be disabled). When setting ticker-tape or autocue, you just specify the parameters for determining the number of measures in each system; the number of systems per nominal page is always 1. You can retain the existing system breaks by choosing the list. In ticker-tape or autocue contexts, all system breaks are converted to page breaks, whatever is in the list.

20.2. Navigation

All the page navigation functions will go to the nominal system page. For example, in a longish piece, if you are on system page 2, say, it will be displayed with system 1 and system 3; if you go to the next system page, system 3 will be displayed with system 2 and system 4. Whenever you go to a page, the selected note will normally be at the beginning of the actual system, not the beginning of the displayed page portal (unless you are at the beginning of the piece).

20.3. Nominal Page Width of a Ticker-Tape Context

The nominal page width of a ticker-tape context is what the user specifies in the layout. This specifies the nominal width of a single *system page*. Typically, this will be appropriate for a single system. The *Systems...* dialog will use the nominal value for determining systems breaks. When a ticker-tape context is displayed, the actual page width used for the display is based on the nominal page plus the length of two staves. It is reduced a little bit further to allow for the fact that reminder clefs and key signatures are not shown within the nominal page, except at the beginning.

20.4. Nominal System Separators in a Ticker-Tape Context

If the *Show Systems Separators* option for a ticker-tape context is turned on (which it is by default), the start of each nominal system is indicated by a small extension of the thin components of the associated bar line above and below the staff; furthermore, the bar number is shown in grey above the first staff. Otherwise there is no discernable boundary between the systems.

Note that in an Autocue context, system separators are shown as normal.

20.5. Page Size & Zooming

In ticker-tape or autocue mode only the first page values for the space above the first staff and below the last staff are used. The values for normal pages are ignored. This is to ensure that the systems remain in a constant position (unless changed by parameters) and, thus, the *ticker-tape* or *autocue* flows smoothly.

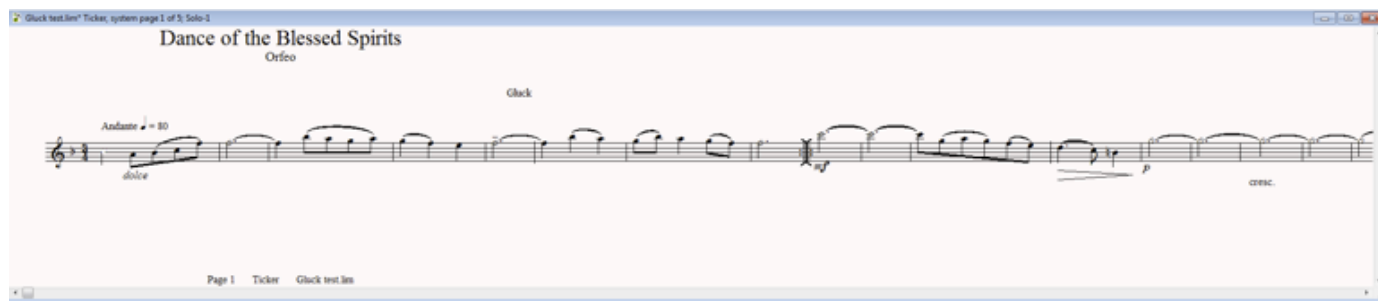
For ticker-tape, it is recommended that the vertical page size is set appropriate to a single system; for autocue, it should be appropriate for three systems.

If the zoom of a ticker-tape context is set to frame/screen width, the zoom is set so that there is a reasonable chance that all nominal system pages will fit on the screen, in Lime Lighter scrolling mode, when the highlighted measure is the first on the system page. Because each actual page comprises three concatenated systems, some systems pages will be expanded from their nominal width and some contracted. If the density is very variable, then inevitably some system pages will be too long and some too short.





20.6. Editing



Editing a Ticker-Tape Context

When a ticker-tape or autocue context is displayed in normal mode, each nominal page comprises three concatenated system pages. The page number is that of the nominal system page. In a ticker-tape context the music appears as one long system, with the nominal page (if not the first or last) in the middle.

In an autocue context, there appear to be three systems, with the nominal page (if not the first or last), being the middle system.

20.7. Annotation Anchoring

When displayed in ticker-tape mode, annotations that are anchored to the page (left, right or centre), or to the staff (left or right) are anchored relative to the beginning or end of the nominal system page within the page portal. For staff anchoring, this is where the bar-line is; for page anchoring this is adjusted by the distance of the staff from the page edge, so they are approximately where they would be if displayed normally.

Annotations designated as appearing on each system or page, will only appear at the beginning of a displayed page, never at the nominal system boundaries within it.

20.8. Printing

Ticker-tape and autocue contexts cannot, currently, be printed, because each system would end up being printed three times and, moreover, ticker-tape contexts would normally be too wide to be useful!





21. Miscellaneous Items

These are in no particular order.

21.1. About Lime

Information about Lime can be found:

On Macs: In the 'Lime Menu', just to the right of the Apple menu.

On Windows: In the system menu, brought up by clicking the Lime icon at the top left of the main Lime frame window (or pressing Alt-Space).

21.2. Edit->Clear

Several new options have been added:

Kill Annotation

Allows selected annotations to be completely removed and, if any were created as a result of "any-Alt-Click", the original graphic will be restored. Unlike clear (delete), killing an annotation has no side effects (such as changing volume levels if the annotation is a dynamic).

If applied to selected notes ("Kill Note's Annotations"), all annotations associated with the selected notes will be killed.

Restore Alt Annotations

For all selected notes, any annotations created following the use of "any-Alt-Click" will be killed and the original graphic will be restored.

If it is selected with the shift-key (or, on Macs, option-key) depressed, then for all selected notes, any annotations created following the use of "any-Alt-Click" and then deleted will be made visible again.

All Note head Coloring

This can be used as a quick way of removing all colours from note heads in the whole piece. There is an option to clear colours from clefs, this is to cater for a rare bug in an older version of Lime (which spuriously left key signatures coloured) and for a planned facility to colour key signatures.

Hidden Annos on Pruned Rests

When a piece is formatted, if a staff chord includes notes and rests or more than one rest in the same staff direction, Lime will '*prune*' unnecessary rests and not show them. In older versions of Lime, any annotations on such '*pruned*' rests were ignored and not seen. When an older file is upgraded, Lime attempts to identify such annotations and explicitly hide them (but cannot always be successful). This clear option allows one to unhide or delete (use the check-box) any such annotations.

Redundant Duplicate Annotations

Removes all annotations that have a duplicate assigned to the same note. True duplicates caused by superfluous duplication with "At Each System" and "On Each Page" are removed when a piece is read. This option will clear any additional redundant annotations and includes provides the option to identical lines and curves.





Substitute All Uninstalled Fonts

When a piece is read, if an annotation uses fonts that are not installed on your system, a substitute is used for display purposes. However, unless the annotation is edited, if the piece is saved the original font identifier is still used and, if the piece is opened on a machine with the font installed, it will be used. This clear option replaces the original font identifier with the one used for substitution. There will be no discernable change on your system, but it will mean that the substitute will be permanent even if the original font is installed.

21.3. Dialogs

21.3.1. Entering Numbers in Dialogs

In all Lime dialogs, a decimal number may be entered as a (heavy) fraction in the form *n/d*.

21.3.2. Tri-State Check-boxes

In some cases a single check-box is used for 3 possible values, true, partially true and false. The exact meanings will, of course be dialog dependent. Clicking the box will cycle through the possible values.

In some cases, the prevailing conditions are such that only 2 possible values are actually possible, in which case only those will be selected. Sometimes the partially selected state is used to indicate that it has been set by default, with default associated values, and the user hasn't yet changed it.

21.4. Note Entry

21.4.1. Grace Notes

When using step mode note entry, when a note length is hit on the piano window, it no longer clears the Grace status, unless the shift key is down.

21.4.1.1. *Nachschlag* Grace Notes

When chords are inserted in step mode, there is special handling if the chord is specified as a *Nachschlag* grace note:

1. If the insertion-point is a normal note, the *Nachschlag* is placed after any other *Nachschlags* already on the note (of, if none, after the note itself). The insertion-point remains the original note so that, if any further chords are added, the same procedure is followed.
2. If the insertion-point is itself a *Nachschlag*, it (and, if necessary any following ones) will be replaced by the new chord (as is normal with step mode entry). The new insertion-point will be the next *Nachschlag* note, if there is one, otherwise it will revert to the original note in the sequence.
3. If the insertion-point is a grace note (but not a *Nachschlag*), the *Nachschlag* that is being inserted is converted to a simple grace note and the insertion-point is set to the newly entered chord.
4. If the chord being entered is a grace note, but not a *Nachschlag*, and the insertion-point is a *Nachschlag*, the new note is placed before the insertion-point *Nachschlag* and becomes the new insertion-point.

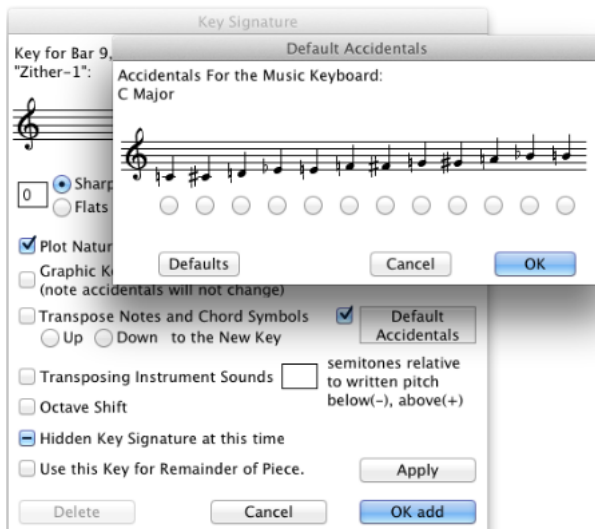




21.4.2. Default Accidentals

When new notes/chords are entered, if an accidental is needed for a note, it is chosen from the prevailing default accidentals. Each time there is a key change, the prevailing default accidentals are reset based on the key. They may be temporarily changed using the *Note->Default Accidentals...* dialog.

As of Lime 9.16.7, default accidentals for note entry can be associated with a key signature. This means that, if your required defaults are different from the normal ones associated with the key, they can be specified permanently. This is done using the *Symbol->Key Signature...* dialog to insert a new key signature or modify an existing one.



Key Signature Dialog showing the Default Accidentals pop-up

In the *Key Signature* dialog, if the *Default Accidentals* check box is checked, the specified set is maintained permanently with the key signature. They will be used as the default, whenever it is the prevailing signature. The button invokes a *Default Accidentals* pop-up.

A common use of key signature default accidentals is to simplify the entry of music when the key modulates but the key signature remains the same. For which purpose, there is the ability to hide the signature (see section 15.4)

21.4.3. Transposing Instruments

There are a number of accessible facilities to simplify entering notes and composing for transposing instruments (e.g. a clarinet in B flat). A transposing instrument is set in the key signature for the part and specifies the number of semitones (half steps) the sound is transposed from the written notation.

A common approach is to enter the music with transposing turned off. Once it is all entered, the transposing instrument can be set in the first key signature of the part and there is a facility to propagate the transposition through any changes of key (see 21.5.3).

The disadvantage of this simple method is if there are other parts in the piece and one wants to hear the parts together as the music is being entered. For this case, the simplest approach is to enter the music as it will sound and then transpose the whole part, setting the required transposing instrument (see section 21.6).





In addition, there is, as of Lime 9.16.9, a preference option ("*Notate as Played on Piano*" in General Preferences) to govern the sound and notated pitch of notes entered into a transposing part. Currently this is a 3-way check-box; the options are:

- Unchecked,** The piano sounds as played, and the notation will be transposed so that, when played back, it will sound the same. This is how it always was prior to 9.16.9.
- Fully Checked,** The piano sounds as played, and the notation will be the same, but when played back, it will be transposed.
- Partially Checked,** The notation will be exactly as played, but the piano sounds as it will be transposed when played back.

These preferences affect all methods of entering notes for a part, using the piano window keyboard or short cuts, when using an external Midi keyboard in step mode or when recording. Note that, if the part is not a transposing part, then all options produce the same WYSIWYG result. When importing from a Midi file into a transposing part, there is an option to notate as the Midi sounds, rather than transposing it.

21.4.4. Replacing Tuplets by Other Tuplets

When entering notes, if the length of the number of notes that must be deleted to accommodate a new note is greater than that note, additional rests are created. Normally Lime will try to ensure that they terminate cleanly at a 32nd (demi-semi-quaver) boundary or at a compatible tuplet boundary. If tuplets are being replaced by incompatible tuplets, this can result in a lot of notes being deleted, but it does avoid needing to create spurious tuplets to fill the gap.

If, and only if, note entry mode is entered using the *Voice->Note Entry* menu option (*Cmd/Ctrl-N*) with the **Shift** key, Lime will terminate the deletion as soon as a viable set of rests can be created, using any tuplets. This esoteric feature might be useful when deliberately fiddling with incompatible tuplets and too much is getting deleted. When active, this 'tight' mode is indicated by the menu option being partially ticked. When **Shift** is down, clicking the menu option cycles through the modes.

21.5. Key Signature Dialog

21.5.1. Short Cut to Set Sharps or Flats

When entering the required number of sharps or flats into the Key Signature dialog, if the first character typed is '+' or '-', it acts as a short cut to set whether it is sharps (+) or flats (-) automatically. The '+' or '-' is then removed.

21.5.2. Default Accidentals

Default Accidentals can be associated with a key signature; see section 21.4.2.

21.5.3. Transposing Instrument

If the *Transposing Instrument* check box is checked, then the value in the associated edit box will be applied, provided it is valid. It is the number of semitones (half-steps) that play-back will be, relative to the notated notes. For example, to set up a part for a B flat instrument, one would set the value to -2.

If the check box is fully checked, the effect will be propagated to the end of the piece, whether or not "*Use This Key for Remainder of Piece*" is activated. If it is only partially checked, it will only be applied until the next key signature change.





21.5.4. Octave Shifted Play-Back

In addition to setting a transposing instrument, it is possible to temporarily set octave-shifted play-back, such as 8^{va} and 8^{vb}. Place a hidden key signature before and after the required section and specify the required octave shift:

- *before* specify +1 for 8^{va}, -1 for 8^{vb}, etc (up to ± 3). This adds a temporary octave shift over and above any existing transposition;
- *after* revert the transposing to what it was before the section.

21.5.5. Deleting Key Signatures

As with other symbols, key signatures can be deleted. However this is not recommended, except in special cases. Normally one should use C major or A minor, which have no accidentals.

If one is deleted in the piece, the signature reverts to the previous one. It is possible to delete the first key signature for a staff and, moreover, if "*Use This Key for Remainder of Piece*" is activated, all key signatures in the piece.

Its one characteristic is that it cannot be transposed, except by adding a signature, transposing it, then deleting it again! A part with no key signatures could be useful for pitch-less percussion, to avoid any possibility if being transposed.

21.5.6. Showing Naturals Before Bar Line

As of Lime 9.17, there is an option to plot naturals at key change before any preceding bar.

21.6. Clefs 8^{va} and 8^{vb}

As of Lime 9.17, all clefs can be specified as up or down one or two octaves. These are shown by an ⁸ or a ¹⁵ above or below the clef symbols. The Vocal Tenor clef (treble clef down an octave) is an example and has always been supported by Lime. Since 9.17, all clefs can be modified.



21.7. Transposing Parts

From Lime 9.16.9, there is a facility (currently, the menu item *Voices->Transpose Part...*) to transpose one or more whole parts.

Whilst the Key Signature dialog allows one to set a key signature and transpose the notes following it into the new signature, such transposition always stop at the next explicit signature. This means that if a piece changes keys and one wishes to retain the changes, one would have to do the transposing key signature by key signature.

The *Transpose Part...* facility will transpose the specified parts, including all their key signatures and any chord symbols. It always operates from the selected position until the end of the piece. Normally one will select the whole piece from the beginning, but it is possible to do it from a key signature change.

Apart from specifying the amount to transpose in semitones (half-steps), there is an option to set the parts' *Transposing Instrument* values in their key signatures to compensate for the transposition, so the result will still sound the same. This can be particularly useful for modifying music for transposing instruments. Create the part as it would be for a normal instrument (e.g. by copying another part), then transpose the whole thing.





21.8. Showing Tuplet Numbers

By default, the number of notes in a tuplet group is shown in the middle of the group, with no bracket, as shown in the following example. If you want tuplet brackets, you should hide the tuplet numbers and draw a *tuplet bracket* line annotation.



Tuplet Numbers

If, for any reason, a tuplet group is incomplete (they are both complete in the example above), then the nominal number is still shown²⁹, subject to the *Show Tuplet Numbers* piece option setting.

Automatic showing of the tuplet numbers number can be controlled in one of two ways:

- By using the *Note->Hide Tuplet* menu item for one or more notes.
- Globally for the context using the *Show Tuplet Numbers* option in the *File->Options for Piece* dialog. If this is fully checked, then all tuplet numbers will be shown, except for those individually disabled. If it is partly checked then only numbers for complete groups will be shown (this is for compatibility with previous versions of Lime). If tuplets are being shown and a tuplet extends over a system break, it will be shown both before and after the system break.

Tuplets can also be shown by explicitly drawing a tuplet bracket (line) on one of the notes, usually the first. Optionally this can include the ratio.

21.9. Parenthesized Accidentals

As of Lime 9.17, it is possible to specify that, if a note's accidental is shown, it will be in parentheses. It is controlled by the little parentheses () button to the right of the accidentals in the Piano Window.

21.10. Piano Window

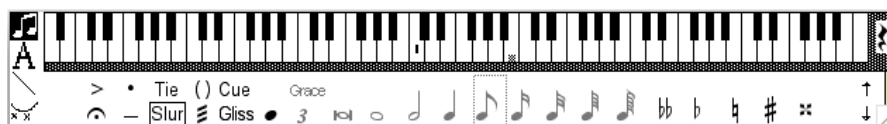
The Piano Window (tool-box) is resizable on both Windows and Macs.

Using the *Window->Piano Zoom...* menu, standard sizes from 100% to 200% may be set. The default is 125% because that is better suited to modern screen sizes and resolution. When the menu option is used, the window is always repositioned to its default position. Other sizes may be set by using the mouse (grow-box at the bottom right on Macs, or any of the window borders on Windows).

The piano window may be moved to a new position using the drag box - the box on the right of the window; on Macs it is light green; on Windows it has a light grey pattern.

²⁹ Before 9.15.9 the nominal number was never shown for an incomplete tuplet group. When pre-9.14 or earlier pieces are imported, the option will be set to *not* show incomplete tuplet numbers.





Mac Piano Window



Windows Piano Window

There is an option, in the Window->Piano Zoom... menu, to minimize the piano window. This can be useful when all one wants to do is read the score. When hearing with tracking or when low-vision scrolling, the piano window is automatically minimized and restored to its original state afterwards.

On Windows, apart from the icons on the left and keyboard itself, which are always black and white the piano window's buttons and background follow the user's Windows' theme.

21.10.1. Piano Window Mode

The 4 buttons on the left of the piano window identify the current mode of music editing.

- Music mode for viewing and editing notes;
- Text mode for creating and editing text annotations;
- Line mode for creating or modifying Lines, etc;
- Curve mode for creating and modifying curves.

The icon of the selected mode is inverted. In the examples, music mode is selected.

Clicking one of the icons will change the current mode of music editing to the associated mode. If already in the required mode and a single item (note or annotation) associated with the mode is selected, it will be scrolled into view, if not already visible.

If the SHIFT key is used when clicking *Text mode* and no annotation is selected, Lime will not automatically enter text edit mode for a new annotation.

21.10.2. Piano Window Music Mode Buttons and Status Indicators

In Music Mode, the two rows of buttons below the keyboard show the status of the currently selected note(s) and many of them may be used to change the status. Those that are inactive are shown disabled (typically, light grey). A button is framed (has a rectangle drawn round it) if all the selected notes have that characteristic; it is shown with a dotted background if only some of the notes have that characteristic.

In normal editing mode, the note duration buttons (and the grace button) are disabled (grey, or whatever is configured for the disabled colour on Windows); however they show the status of the currently selected note(s). In addition, a small, grey mark is placed on one or more piano keys, corresponding to the selected note or notes.

In note entry mode, all buttons are enabled and are used to set the characteristics of the next note to be entered.

In duration editing mode (*Edit->Enable Duration Edits*), the note duration buttons (and the Kill button), which can be used to change the duration of the selected note(s) are shown in red (whatever the Windows' theme), indicating that care must be taken when using them.





21.10.3. NUM-Lock Indicator

The drag box on the piano window is used to indicate whether the keyboard is in NUM-Lock state. NUM-Lock is indicated on the piano window by the drag box being darker (darker green on Macs, darker grey on Windows).

If you wish to use the number-pad arrow keys, then it is important that NUM-Lock is off. Similarly if you wish to use the numbers on the number pad (typically for piano window short cuts), then NUM-Lock must be on.

On most keyboards³⁰, NUM-Lock is toggled using the key above the '7' on the number-pad. On Macs the SHIFT-key must be used as well³¹.

21.10.4. Piano Window Zoom menu

The *Window->Piano Window Zoom* sub-menu can be used to change not only the size of the piano window but also whether it is shown at all.

Windows The *Minimize* and *Float on Top* items are complementary.

Minimize Minimizes the piano window;

Float on Top restores it.

Macintosh The *Minimize* and *Float on Top* items are toggles.

Minimize Toggles Minimization of the piano window;

Float on Top Toggles whether the piano window is always in front of all music windows. Clicking it also rescinds any minimization.

21.11. Rulers

If rulers are enabled (in *Edit->Preferences*), a click in the horizontal or vertical ruler, it will draw a guide line; you have to click the ruler and drag it away to get rid of it.

In addition, if the **Alt** (Option) and **Shift** keys are pressed (*but not the Command/Ctrl key*) while moving the mouse about, a full page cross-hair will be shown³².

21.12. Low Resolution Rendering

21.12.1. Printing

If the Shift-key is down when printing is selected, an alternative ("low") resolution method is used.

Windows: The alternative "low resolution" is simply a slightly different rounding algorithm. In most cases, it makes little or no difference.

³⁰ On some keyboards (notably the Dell KM714), NUM-Lock is permanently on & can't be changed.

³¹ On Macs, NUM-Lock is controlled within Lime and always starts in the *off* state.

³² Prior to 9.16 this was Command/Ctrl + Shift, but this was incompatible with the use of Shift to constrain drawing and dragging to the horizontal or vertical.





Macs: By default Macs use very high (10x) resolution when printing and specify a scaling of 10% (over and above any user scaling). If saving to a PDF file, the result can be quite large. The “low resolution” option uses 5x resolution and, thus, results in smaller PDF files.

21.12.2. Copy Rectangle

By default, Copy Rectangle will copy at high resolution (5x), which will need down-sizing when pasting; this is done automatically by word processors, such as Word. If the shift-key is used (“low resolution”), then an exact copy of the rectangle on the screen will be made at the current screen resolution.

Note that nowadays, particularly on Macs, copy rectangle is deprecated, because the equivalent can be achieved by printing to a PDF and using PDF tools to copy selected portions.

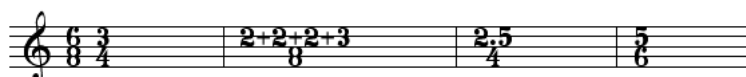
21.13. Time Signatures

21.13.1. Advanced Time Signatures

Lime supports *additive* and *fractional* time signatures as well as *alternating* (sometimes known as interchangeable) ones. Check the *advanced alternation* checkbox for alternating ones. For additive or fractional ones, enter the required combination. Up to 4 additive beat counts may be specified. For fractional ones, enter the fraction as a decimal number; only n.5, n.25 and n.125 are supported.

If the Compound option is selected for fractional or additive signatures, then the actual display is simple, but the metronome will follow the underlying rhythm. For example, $\frac{6}{8}$ will normally be the equivalent of $\frac{3+3}{8}$ (2 beats in the bar). For common compound signatures, such as $\frac{6}{8}$, the conversion will be done automatically when compound time is selected.

Irrational time signatures (e.g. 5/6) are also supported, provided it is possible to create the necessary tuplets to fill the measures.



Alternating, Additive, Fractional, Irrational,
Time signatures

Complete section on advanced time signatures

21.13.2. Placement

Time signatures are normally placed visibly on the staff to which they refer. The time signature dialog includes a placement option which, in addition to placing normally on the staff, allows it to be placed above or below the staff, or above or below the whole system, or be hidden completely.

21.13.3. Polytemporal

If a time signature is *Polytemporal*, the associated *Polytempo Duration Multiplier* changes the duration and spacing of every note after this time signature. In essence, the multiplier is applied to the effective note length. If, for example, the multiplier is 1.5, then a quarter note (crotchet) will take the same time as a dotted quarter note on a part without a multiplier.





Because it is a very, very rarely used feature, the polytemporal multiplier is only shown and is only active if the *Polytemporal* option is checked. The actual entry of the multiplier uses a separate dialog, that allows it to be entered either as a decimal number or, more usefully, as a (heavy) fraction or the product of two fractions - see section 17.4.

21.14. Measure Insertion and Deletion

The *Page->Insert Measures...* dialog includes an option to insert a pick-up measure. If this is selected, a single, unnumbered pick-up measure is inserted; if this is at the beginning of the piece, it will be measure 0.

Both the *Page->Insert Measures...* dialog and the dialog for creation of new pieces, include an **Advanced** button, which enables a complex time signature to be specified (such as Compound, Alternating, Fractional, or Polymetric). If the complex signature involves different measure lengths, these will be inserted correctly. For inserting measures, there is an option to insert the time signature at the beginning of the insertion; when creating a new piece, the default is to insert the time signature at the beginning.

When new measures are inserted, Lime will try to retain the original selection if it is still visible on the current page. If it is not on the current page, Lime will try to select the first rest of the new measures; if that is on the current page, the last note before the insertion (and before end of page) will be selected.

When measures are deleted, Lime will try to select the first note after the deleted measures if it is visible on the current page, otherwise the note before will be selected.

21.15. Symbol Placement at System Breaks

21.15.1. Bulk Changes - Systems Option

When using the *Page->Systems...* facility, the *Clef/Key Changes at end of systems* option can be used to specify whether, when new system breaks are inserted, clef and key changes are put at the end of the previous system. The default, partially checked, means they will be inserted unless there was already a system break (*in any context*) at that point without them; if fully checked, changes will always be inserted; if unchecked, they will not be inserted.

21.15.2. Manually Splitting Systems

When the *Page->Split System* facility is used, If there is a change in Clef, Key or Time Signature at a new system break, unless there was already a system break (*in any context*) at that point, the user will be asked whether the symbol(s) should be placed before or after the break.

21.15.3. Clef, Key Signature and Time Signature Dialogs

The *Clef*, *Key Signature* and *Time Signature* Dialogs provide options for controlling placement of these symbols at the beginning or end of a system.

Unless suppressed, the prevailing clef and key signature is shown at the beginning of every system; this is the normal convention. Time signatures are normally not shown at the beginning of systems, except when there is a change (including at the beginning of the previous system). When there is no change and such symbols are shown at the beginning of a system, they are known as '*Reminder*' symbols.





If the insertion point is at the beginning of a system, if there is not already a symbol change there, Lime will offer the option of inserting the symbol change at the end of the previous system. It is common to show such a change at the end of a system, and repeat it at the beginning of the next.

Occasionally, it may not be desirable to put a change at the end of a system (for example, when starting a new logical section). It is possible, therefore, to move symbols at the end of a system to the beginning of the next (providing there is not already one there). To do this, bring up the appropriate dialog on a Note before the end of the system, select '*At End of System*', and then specify '*Move to Next System*'. There is also an option to move a symbol at the beginning of a system to the end of the previous one - bring up the appropriate dialog on a Note at the beginning of a system, select '*Before Note*', and then specify '*Move to Previous System*'.

21.16. Next, Previous Buttons and Apply Buttons in Symbol Dialogs

The *Clef*, *Key Signature* and *Time Signature* Dialogs provide the ability to navigate to the next or previous "change" of the symbol. The reminder key signatures and clefs at the beginning of systems are not considered a "change". The *Previous* and *Next* buttons are enabled if there is a previous or next change.

21.16.1. Apply

The '*Apply*' button applies outstanding changes but stays within the dialog, allowing further changes and use of the *previous* and *next* buttons.

21.16.2. Previous and Next

The '*Previous*' and '*Next*' *change* buttons go to the previous and next "change" of the symbol in the current context (on different page if necessary), cancelling any unapplied changes.

The Next button skips over a symbol at the end of the current system; this is because it can be edited directly using the *at end of system* checkbox.

21.17. Saving

Whenever a piece is saved, it is first saved to a temporary (unique) file in the target directory, which is then renamed to the required file name. This means that, if there is a problem while writing a file, the original file will not be corrupted. Error messages may sometimes refer to this temporary file.

Though most potential problems (such as read-only) are detected before a file is written, it is possible for there to be an error when trying to rename the temporary file. If this happens, the temporary file will be left in the target directory for the user to manually move or rename (or delete).

By default, a piece may be saved even if it is unchanged. Whenever a piece is saved, its status, such as current page and context is saved with it. There is a preference option to use Save As... if you try to save an unchanged piece.

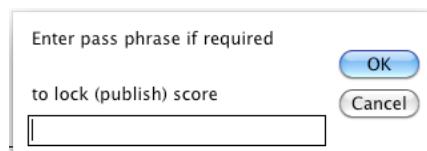




21.17.1. Locking (Publishing) Scores

It is possible to lock (often called 'publish') a score so that nothing in it can be changed, even if it is saved as another file name.

A score may be locked (published) by choosing the *Save as Locked...* item from the File menu. You will then be asked to enter a 'pass phrase', which, if not empty, must be used if the score needs to be unlocked.



If the pass phrase is empty (or all spaces), then nothing is needed to unlock it. Note that any file 'published' prior to version 9.15 has no pass phrase and can be unlocked easily.

21.17.2. Undo After Save

Lime retains memory of changes for two saves of the piece after the changes were made. If you attempt to undo to a state before the last save, you will be asked whether to proceed (if you give the go-ahead, you will not be asked again).

21.17.3. File Size

Note that the sizes of Lime files on disk are approximately twice the size of what they used to be with Lime 9.0x. This is to support new features and much larger zoom sizes.

Prior to Lime 9.16.6 there was an obscure bug in Lime when systems were restructured that had annotations duplicated *at each system*, which were in *all contexts*. Such annotations were duplicated exponentially, resulting in some cases in extremely large files. When a file is read, Lime 9.16.6 and later silently remove such duplication. If there were lots of duplicates, when the file is saved, it can be significantly smaller.

21.18. Go To Bar and Go To Page

Both the *Go To Bar...* (*Ctrl/Command+G*) and *Go To Page...* options in the Page menu, bring up a touch screen friendly dialog for entering the required Bar or Page number. The dialogs default to the specified option, but there is an alternative button for going to the other option.

Bar or page number 0 (zero) will always go to the beginning of the piece, whether or not there is a pick-up measure at the beginning.

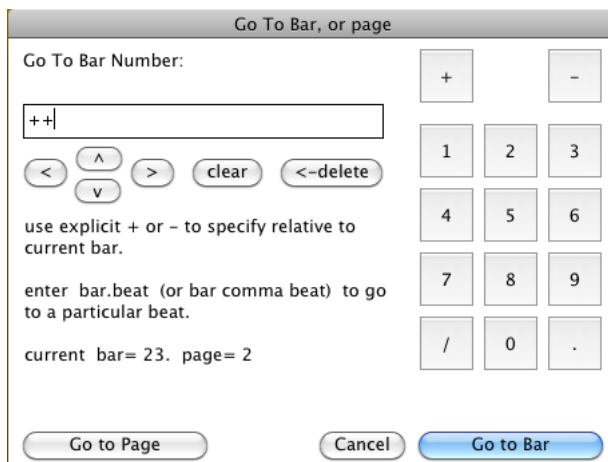




On Windows:

In the *Go To Bar...* dialog, you can specify page by using alt-P instead of return.

In the *Go To Page...* dialog, you can specify bar by using alt-B instead of return.



Go To Bar Dialog

When used from the Lime Lighter music stand, an enlarged version is used and it includes a *Stop* button to exit Lime Lighter tracking.

The dialogs provide a numeric keypad for use with a mouse or touch screen to enter the numbers as well as navigation buttons to move the cursor.

As well as allowing entering a specific Bar or Page number, the dialog supports specification of relative Bar or Page numbers, by explicitly putting + (plus) or - (minus) before the number. Alternatively one can use a string of plusses or minuses; the relative value is the number in the string (the example above means go forward 2 bars to bar 25).

21.18.1. Going to Beginning of Current Page

If used with the Shift key, both the *Go To Bar...* and *Go To Page...* options go to the first measure of the current page.

21.18.2. Going to First or Last Page

Use of the shift key with previous or next page menu items (or with Command/Ctrl-[and Command/Ctrl-]) will go to the first measure on the first page or the last measure on the last page.

21.19. Go Back to last selection

The *Go Back to last selection* option in the *Note* menu (or *Ctrl/Command+Shift+Space*), will select the note that was previously selected before the current one, changing pages if necessary. Successive invocations will toggle the selection between two notes.

If the music has been edited, it is possible that the previous selection is no longer available, in which case the option will not be available.





21.20. Merging Staves

There is a facility for merging staves throughout the piece. All notes on one staff, throughout the piece (in current context) can be merged with either the one below or one above. This works better than the Voices/Parts dialog because it caters for voices stopping and starting.

To use this facility, select any single note on a staff and choose the *Voices->Merge Staff...* menu item. You will be asked whether merging should be upwards (with previous staff) or downwards (with next staff).

Merging staves replaces *Staff Drag* as a way of printing two staves together.

21.21. Rhythm Errors and Voice Length Consistency Checks

By default, whenever Lime reads a piece, it checks that all voices are approximately the same length when it comes to play-back. If for some reason, you have a piece, which always fails this test, you can set the piece options to omit this test when reading the file. Currently this can only be set partially (to allow for an expanded option in the future).

21.21.1. Next Rhythm Error

The *Page->Next Rhythm Error* menu item will go to the next bar whose length is inconsistent with the time signature. This can be useful after importing (or opening) a file with known timing errors.

In normal mode this option will just search the currently selected voice, from the current position. If the SHIFT key is pressed when the menu item is selected, all subsequent voices will also be searched from the beginning of the piece.

21.22. Don't Share Note Head

The *Note->Don't Share Note Head* option in the Note menu, allows you to specify that, if there are multiple voices on a staff, a note's note head should not be shared with that of a note in another voice.

Normal layout with shared note heads



Layout with Don't Share Note Heads



In the example above, the stem up quavers (blue eighth notes) have "Don't Share Note Head" set.

21.23. Switching Contexts

The short cut Ctrl/Cmd-semicolon (;) can be used to switch contexts quickly to the previous one viewed. Ctrl/Cmd-shift-semicolon will go to the score context or, if already there, to the last context in the list.





21.24. Select All

There is now a 'Select All' sub-menu under the Edit menu. This allows one to select all notes in a voice, part, on a staff or in the whole piece.

21.25. Beaming over Rests

By default, when a beam goes over a **visible** rest, Lime 9.16 (9.16.3) will put a small stem (known as a *stemlet*) on the beam where the rest is. The length of this stemlet is governed by the *stemlet* parameter (under BEAMING). If no stemlet is wanted, then this can be set to zero.

There is a piece option to default the *stemlet* parameter to 0. For backwards compatibility, to avoid the changes in beam positioning the stemlets require, this is set by default for any file created before 9.16.5.

Prior to 9.16, not only were there no stemlets, but also partial sub-beams were not put on rests, because they would have simply floated. After 9.16, even if the stemlet length is zero, it does join the sub-beams so that they do not float. If you really want no sub-beams on a rest, like it was before 9.16, then the "no stem" attribute can be used to specify that no stemlet at all should be used and partial sub-beams should be omitted on the rest.

21.26. Multi-Length Staff Chords

In general it is recommended that, when a staff contains more than one musical voice, each voice is notated as a separate Lime voice. In some cases, where a staff chord contains more than two different note lengths, it may be necessary to have one or more additional voices. The use of multiple voices allows Lime to format multi-length chords properly (using a layering technique). The following example (from a piece by Purcell), uses three voices (coloured red, blue and black):



Multi-length staff chord, using 3 voices

Prior to version 9.16.5, if there were insufficient voices in a multi-length staff chord to format it properly, Lime would simply discard and not notate longer notes, so that only one length would appear in each stem direction. For example, if a chord containing an eighth (quaver), a quarter (crotchet) and a half (minim) is created using the 'add' facility on a single voice, it is not possible to have just one length in each stem direction, so the half note would have been discarded. Furthermore, if all notes were forced to the same stem direction, both the half and the dotted quarter would have been discarded.

As of version 9.16.5, such notes are no longer discarded; all will be printed. In the following example, all notes are on the same Lime voice (prior to 9.16.5, the half note would not have appeared):



Multi-length staff chord, using 1 voice





This can, of course, result in some confusion as to what the actual length of a note is. In the above example, it is obvious because the note heads are different. However, if the note heads are the same, it may not be obvious. In these cases, unless *continue primary beaming* is set, by default primary beaming will be ended before a multi-length staff chord; also unless *continue secondary beaming* is set, by default partial secondary beams will be used on the multi-voice staff chords. This is shown in the following example; the red notes are eighths and the blue ones are sixteenths (and, for the purposes of the example, both red and blue notes are forced stem up):



Incomplete beaming is an indication of multi-length staff chords

Whatever technique is used, care is needed to achieve the desired stem directions and beaming. Also *chord drag...* may be needed to get the desired horizontal positioning.

Note that a superfluous rest is normally pruned from a chord with notes unless it is the shortest length object in the chord, when it is important to the rhythm.

21.26.1. Accumulated Arpeggios

An example of the use of multi-length staff chord is an "Accumulated Arpeggio":



Simple Accumulated Arpeggio

As of Lime 9.16.5, this can be entered all on one voice, by first entering the arpeggio components and the final chord, then adding (select *Add* in the piano window) the tied components, namely an 1/8th G to the 1/16th G, then another 1/16th B; in both cases make them stem down.





21.27. Braces

As of Lime 9.16.6, curve based braces at the beginning of systems are supported. This is in addition to the square braces used when the music font is Marl/Tufa and the large, curvy braces used when the music font is Sonata.



Square Marl/Tufa Braces



Very curvy Sonata Braces

Braces are controlled by a parameter under *BRACKETS and BRACES*:

Brace shape: 1=plain 2=fat 3=square 4=Sonata 0=1/Marl 2/Sonata, <0=old

- <0 Square braces if Marl/Tufa music font, very curvy ones if Sonata music font.
- 0 Plain curves if Marl/Tufa music font, fattened curves if Sonata music font.
- 1 Plain curves.
- 2 Fattened curves.
- 3 Traditional old square braces, based on Symbol font characters if possible.
- 4 Traditional old curvy braces, based on Sonata font characters, if possible.



Plain Curve Braces



Fattened Curve Braces

Any negative value (the current default) specifies the traditional square braces if the Music font is Marl/Tufa, and curvy braces if the music Font is Sonata. The other parameters that control the details of the traditional braces are deprecated.

It should be noted that the square braces use characters in the Symbol font, which must of course be installed to allow the braces to be drawn. Use of curve-based braces is recommended, particularly on Windows, as they do not require any font to be installed. Furthermore, there seems to be some problems on older Macintoshes reading PDFs created on Windows using square brackets (fixed in OSX 10.10). There is also a minor problem with square braces with large staff sizes.

Currently, on Macs, the curved braces are a little jagged on the screen, though OK when printed. Until this is rectified, the default remains the traditional, font-based braces.





21.28. Midi File Export

Any Lime piece can be exported to a MIDI file. The format will be type 0 (i.e. one track with all channels contained therein). As far as possible Lime will export time signatures, tempos and annotations. However, it must be recognized that a MIDI file is primarily designed for hearing, not for exporting a score. If you plan to import the resultant MIDI file into another program, please note:

- Midi only supports time signatures with a denominator that is a power of 2. It is not possible, therefore, to export irrational time signatures accurately. Lime will do a best approximation, within the constraints of the MIDI format.
- The maximum value of a numerator in a time signature is 127.
- Lime constrains the time signature denominator to 128, because that is the limit most MIDI importers will accept without complaining (though many (most?) programs limit the denominator to a 64th note when it comes to an actual score).

21.29. Midi File Import

When importing from a Midi file into a transposing part, there is an option to notate as the Midi sounds, rather than transposing it (see section 21.5.3).

Note that, for type 1 MIDI files, importing the timing track 0 is inhibited.

21.30. Lime Console

For certain operations³³, Lime will write log and status information to a 'console' window, which will usually be shown automatically.

On both Macintosh and Windows, there are 'Hide' and 'Show' Lime Console items in the Window menu.

21.31. Navigation when Inspecting Internal Fields

The *Note->Internal Fields* dialog includes four buttons to enable navigation within the page while staying within the dialog. Each button is equivalent to the *Cancel* button, followed by a move to the appropriate note, followed by re-invocation of the dialog. The four buttons correspond to the left, up, down and right arrow keys, namely:

- < move to previous note
- ^ move to previous voice
- v move to next voice
- > move to next note

If the move is not possible (e.g. you are already on the last note in a voice on the page, or the voice stops printing), the system beep will be sounded and the dialog will remain on the same note.

Note that these arrow keys do NOT save any changes you might have made. If you want to apply a change before moving, use the *Apply* button.

³³ Currently only the Batch... facility uses the console





21.32. Bar Lines Through First Chord (advanced feature)

Some types of modern graphical scores use bar lines through the first chord of each measure. Lime (9.16.4 and later) supports this to a limited extent, with the option to place bar lines through the centre of the first chord in the measure or to the left of the leftmost note head in the first chord. This is done by using a negative value for the BAR LINES parameter, "*space left between bar line and next object*".

When positive, this parameter applies to whatever the next object after the bar line is, note, time signature, key signature, etc, and is scaled to the staff size. When negative, no space is left, no scaling is done and the bar line will be aligned to the first chord in the following measure (if not a whole- or multi-measure rest), as follows³⁴:

- 1: Align bar line to the left edge of the leftmost note head of the first system chord or grace note.
- 2: Align bar line to the left edge of the leftmost note head of the first system chord, ignoring grace notes.
- 3: Align bar line to the centre of the first system chord or grace note.
- 4: Align bar line to the centre of the first system chord, ignoring grace notes.

The different effects are shown in the following figure (in which stems have been retained):

"space left between bar line and next object" parameter

-3: align to 1st chord or grace note centre.

-4: align to 1st chord centre, ignoring grace notes.

-1: align to left of 1st chord or grace note.

-2: align to left of 1st chord, ignoring grace notes.

Aligning Bar Lines with Note Heads

³⁴ Actually any value between -0.1 and -1.0 means -2 and between -1.1 and -2.0 means -2, etc.





Aligning to the centre of a system chord, means the average of all chords in all printing voices on the system that start at that time and have the same parameter setting. If a chord involves note heads on either side of a nominal stem (clustered chords), then the centre will be between the two note heads. If there are such chords with different stem directions, then there will be an additional spread. It may be necessary therefore (particularly if no stems are used), to align stem directions in order to achieve the required vertical alignment.

21.32.1. Experimental

The bar line through 1st chord facility is experimental and will probably not be rendered properly in Lime 9.16.3 or earlier. However, it is worth noting that, providing there are no other objects (or more than 1 accidental) before the first note, the value -3 (or below) has the same effect in previous versions of Lime and is thus vaguely compatible; though there may be a need to reset other parameters, such as *space left between clef and next object*.

It is recognised that the user interface could be improved so that the user can specify one of the special values by function, rather than entering a negative number. It is also recognised that a parameter is needed to control the positioning of clustered chords so that up- and down-stem chords may be vertically aligned

Further study is needed into whether there is a need to do it on a bar by bar basis, or whether there is a need for more options. Currently all values below -2 align to the chord centre, and all even values ignore grace notes; however it is recommended that only values -1 to -4 are used, in case additional options are implemented.

21.33. System Breaks

21.33.1. System Breaks from List

When the *Systems...* option is used to create system breaks with a list of bar numbers, the numbers refer to the **printed** bar numbers in the reference (selected) voice. The printed bar numbers take into account the starting bar number from the *Options for Piece...* and any disjunctions in bar numbering. The numbers in the list are processed sequentially. A bar is found by scanning **forwards** in the piece from the position of the last break; if bar with that number is found, a system break is inserted and then the next number in the list is processed.

If one or more bar numbers are not found, the user will be warned and given the option to ignore them or to cancel the operation.

21.33.2. Ignoring Timing Option

When generating system breaks, Lime tries to insert the breaks at the time of the breaks in the reference voice. Normally, unless polytemporal/polymetric is specified (see section 17.5), if the break is not at a measure boundary on a voice, a warning will be given and the break will be made at the next bar-line. If, however, the *Ignoring Timing* checkbox is checked, breaks will be made at the same measure number in each voice, regardless of timing, and there will be no warnings.

The primary use of this is when setting up or correcting the systems structure following MusicXML or NIFF import. Normally it should be unchecked unless there are known to be uncorrected timing problems in the piece.





21.33.3. Re-Pagination

If the number of systems per page, for any of the options, is set to zero (or blank), then the context will be re-paginated automatically from and including the selected page to optimize use of available space on each page. On each page, after the first system, additional systems are only be added if there is enough room without requiring compression.

21.33.4. Staff Drag

The use of Staff Drag, whilst it is still supported, is very much **not recommended** except in exceptional cases, because the results are difficult to manage.

When systems are split, all staff drags around the break are removed because system and staff separation distances will change and, therefore, the drag amounts are probably no longer applicable.

21.34. Pasting Music

When pasting music, unless the shift key is used, volume levels and reminder accidentals of the pasted notes will be retained. With the shift key:

- If there is no key change, reminder accidentals will be discarded.
- If dynamic markings are set to affect volume levels, they will be recomputed.

21.35. New Note Heads and Accents

Lime 9.16.9 has added a built-in Rhythm/Slash note-head () and an anti-accent ().

Rhythm note-heads and anti-accents are available in 9.16.9 and later. They are recognized by Lime 9.16.8 but not by any version prior to 9.16.8.

21.36. Symbols at End of Staff

There is now a parameter, "*space left after clef/key/time at end of system*", to control the space left after a Time Signature or Key Signature, or Clef (if there is no bar line) at the end of a staff. The parameter is under "*STAFF LINES, MARGINS, NOTE SPACING*".

21.37. Fixing Missing Reminders

If the piece options to show reminder Clefs or Key signatures at the beginning of systems is changed, it is necessary to redo the systems for each of the affected contexts. This may also be necessary if there are missing reminders after importing a piece³⁵.

³⁵ Known to happen occasionally when importing NIFF





The Systems option (*Page->Systems*) will do this on a context by context basis. Choosing either the *fixed system or page start* option, or the *existing systems* option, with *no change*, will add any reminders that are missing (and delete any that are redundant). The *existing systems* option, with *no change*, is recommended if the context is polymetric.

The *Page->Fix Problems->Fix Missing Reminders* option provides the facility to fix reminders quickly, with the option to do it for all contexts.

21.38. Note Stem Flags

A note, whose duration is an eighth note (quaver) or shorter and is not beamed, is normally notated with a stem and a flag, whose number of lines indicate the duration - one line for an eighth note, two for a sixteenth (semi-quaver), etc. These flags are normally curvy using characters from the Marl font. They can be changed using parameters under **Note Flags**; there is one parameter for each curvy stem-up and stem-down flag. Unless you have a special font, the only normal change is to reduce (or increase) their size (the default for all is 12pts, but some people prefer 11pts).

If, however, you prefer straight flags, since 9.16.9 there is a parameter to specify that note stem flags should actually be straight lines, rather than curvy characters.



Curvy Note Stem Flags



Straight Note Stem Flags

The main parameter (the first under **Note Flags**) allows one to specify whether grace notes, normal notes or both are straight.

- 0 All notes' stem flags are curvy (character based) - the default;
- 1 Grace notes' stem flags are straight, but normal notes' are curvy;
- 2 Normal notes' stem flags are straight, but grace notes' are curvy;
- 3 All notes' stem flags are straight.

The characteristics of straight stem flags can be controlled by three additional parameters (also under **Note Flags**), each of which can be specified for normal sized notes and for cue/grace sized notes:

- The line width of each line in a note stem flag;
- The horizontal offset (width) of each line;
- The vertical offset (height) of each line.

The straight note stem flags are available in 9.16.9 and later. They are not recognized by Lime 9.16.8 or earlier, which will simply use the traditional curvy flags.



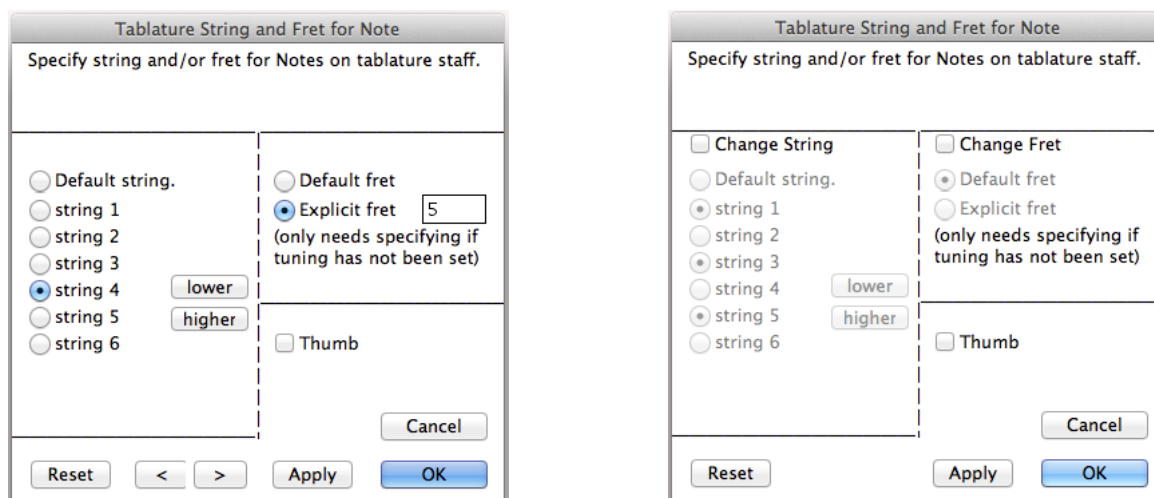


21.39. Guitar Tablature

By default, Lime will automatically assign string and fret based on the prevailing tuning. The default tuning is the standard guitar 6-string tuning. Currently the tuning can be changed using the "*MIDI pitch for string n in guitar tablature*" parameter set under "*Tablature Pitches and Symbols*" (n.b. string 1 is the top string in the tablature).

Before 9.16.7, one could change the string, using the *Note->Tablature String* option. The fret was always assigned automatically by Lime.

In 9.16.7, both string and fret can be specified (or set to the default) using the *Note->Tablature* dialog. If just a single note is selected, string and fret may be simply changed. If, however, more than one note is selected, the user must choose whether to change string or fret, because any change will affect all selected notes. Examples are shown in the following figure:



Single Note

Multiple Notes

Tablature String and Fret Assignment

If the tuning is appropriate, the fret is not actually needed. However it is often set explicitly in music imported from MusicXML and tuning is often not supplied.

The "Thumb" check box is intended to show that the thumb should be used. In the tablature, this is indicated by a very slightly larger and emboldened fret number (in italics if the emboldened preference is used, e.g. for Lime Lighter). This will not be changed unless selected explicitly.

In the dialog:

Apply will apply changes but keep the dialog active.

Reset will discard any changes that have been made to the dialog and revert to the current state (or to last *Apply*).

< and > go to the previous or next note in the selected voice (without making any changes) - if you want changes to be effective, use apply first. These navigation buttons are not available if multiple notes are selected.





21.40. Batch File Processing

The File->Batch... menu offer facilities to batch process more than one file and save the results. The batch capabilities are:

Update	<p>Read and save the selected Lime score files. The files are automatically updated to the current file format. Options are available to:</p> <ul style="list-style-type: none"> • Set or clear 'template' flag. • Set or clear locking (publishing) of a file. • Save or clear graphic data (for Lime GoodFeel). • Set the piece to open on the first page of the <i>Score</i> context. <p>The primary function of Batch Update is to update an old set of Lime files to the latest format.</p>
Export MusicXML	<p>Read and export the selected Lime score files to MusicXML N.b, Lime's exporting of MusicXML is fairly primitive (see section 7.2).</p>
Export Midi	<p>Read and export the selected Lime score files to MIDI files. As with single file export to Midi, there is an option to specify the number of divisions per quarter note; the default, 960, is recommended.</p>

The export capabilities have the option to use the *Score* context. If this option is not chosen, the export will be of the context that was active when the Lime file was last saved.

21.40.1. Destination Specification

All capabilities require the user to specify a folder into which the results will be saved. Normally this will be a different folder to the original files, however the source files' folders can be specified. Saving in the source files' folders is not recommended for batch updating, unless the file name is changed in some way. For this reason there is an option to add an additional period to the name.

To guard against overwriting existing files, there is an option to '*Make ~Backups when Saving*'. If, as is recommended, this is chosen, any output files that already exist, will be renamed with a leading tilda.





21.41. Shift Key Menu Alternatives

The Shift key is often used with menus to modify slightly what the menu actually does. The following is a list of menu items affected by the Shift key.

- *Next* and *Previous Page* become *First* and *Last Page*.
- When Shift is used, *Revert* is available, even if the file is apparently unchanged.
- When Shift is used with *Split System* or *Add to Next System* and the context does not have 'polytemporal parts', the timing tests are not performed and the split or addition will be at the same measure number on each voice (this is not recommended).
- When Shift is used with *Print*:
 - On Macs the resolution is a bit lower, creating smaller PDFs.
 - On Windows a slightly different rounding algorithm is used.
- When Shift is used with *Copy Rectangle*, the copy is at the current screen resolution (zoom).
- When Shift is used with *Paste Text*, the paste will be unformatted; just the text will be pasted, not the style.
- When Shift is used with *Enable Duration Edits*, there will be no warning message.
- When Shift is used with *Create Piece*, it defaults to a grand staff.
- When Shift is used with *Insert Measures*, there will be no warning message if 'all staves' is not chosen.
- When Shift is used with *Copy Music*, all annotations are copied as well.
- When Shift is used with *Paste Music*:
 - volume levels will be recomputed if dynamic markings are set to affect them;
 - reminder accidentals will be discarded.
- When Shift is used with *Extend Line or Curve to Here*, the position of the end will be set to the right of the terminating note head, rather than the left of it.

↳ *Complete list of Shift modifications to menus.*





22. Max OSX Considerations

Note that OSX has a tendency to set the default application for a type/extension to the last application that opened a file of that type implicitly. If that application is removed, OSX will revert to any suitable version it can find, not necessarily the previous version that was being used. This can cause confusion, particularly if the one it found is older. Removing all unneeded versions from the disk (or compressing them and deleting the originals) is a good way to guard against this.

22.1. OSX Version Support

Lime 9.16 has been tested on all systems from 10.4 (Tiger) up to 10.14 (Mojave); it will not work on earlier or later systems. Lime works on both Intel and PowerPC Macs, however support for 10.4 and 10.5 (and PowerPC) is deprecated and may not continue after 9.16.

N.B. Lime is not signed for Mountain Lion, or later. You can bypass the initial Gatekeeper check by right clicking on the Lime App and choosing the '*Open*' command. Once it has been opened one time, Gatekeeper no longer has any control over it.

22.1.1. Warning

Apple has stopped 32 bit applications, such as Lime, working on MacOS 10.15 (Catalina). If you wish to continue using Lime, you can upgrade to 10.14 (Mojave) but do NOT upgrade to Catalina (MacOS 10.15) or later.

22.1.2. Mojave, 10.14

On Mojave the fonts panel no longer works, so the simplified one is always used. Apart from that and not supporting turning off horizontal centering when printing (see 22.2.3), Lime works fine on Mojave.

However Apple have also, unhelpfully, stopped many old USB audio devices working after Mojave. So if you have such a device, check it will work on Mojave; otherwise stick with High Sierra (10.13).

22.1.3. Yosemite, 10.10

A problem has been identified on some versions of Yosemite, which results in strange shadowing of Lime's windows. This is a defect in Yosemite, which may be fixed (if you do not want to upgrade) by turning off shadows. Whilst Yosemite has no option to do this, it can be achieved with an excellent and free application called **ShadowSweeper** from www.IrradiatedSoftware.com/labs. Simply run it to turn shadows off and run it again to turn them back on again. Better still upgrade to El Capitan!

22.2. Printing

22.2.1. Print Quality

Since version 9.15, high quality printing from Lime is available on all platforms.

22.2.2. Post-Script Printing Deprecated

On the old Macintosh computers, including the original Mac systems and OSX PowerPCs (PPC G3, G4 and G5), a Post-Script printing technology was used to overcome graphical capacity limitations. This is no longer available on Intel Macs and is no longer required.





However, on PPC Macs and on Intel Snow Leopard (OSX 10.6 or earlier running Lime in PPC mode under Rosetta), 9.15 still supports the original Post-Script printing technique. On such systems, though it is not recommended, users are given the choice as to whether to use Post-Script or normal graphical printing. There is a preference option to specify the default (the default is not to use PostScript). Post-Script printing is not compatible with full Unicode; if a character cannot be converted to an 8-bit equivalent, it will be printed as a question mark (?).

Note that, on all OSX systems (on Intel or PowerPC, with or without PostScript) high quality PDFs are fully supported. There is no requirement for any PostScript kludges. It should, however, be noted that PDFs produced without PostScript are significantly larger than ones using PostScript or ones produced on Windows.

22.2.3. Print Centering

On Mac OSX (except on Mojave), when printing, there is a print option to centre the print horizontally on the page. This is ON by default as that is what is done on Windows. For completeness, there is also an option to centre vertically, which is OFF by default.

Optionally the defaults for centering may be saved in the context options.

Most printers have a small non-printable margin; if centering is not used, then Lime adds such a margin, so that no additional scaling (over and above what the user has configured) is needed to ensure that the very top (or left) of each page actually gets printed, whether printed directly to a printer or saved to a PDF and then printed.

Without the margin (or centering), in order to ensure that the very top (or left) is printed, it would be necessary to specify 'scale to fit paper size' every time when printing. This also applies to printing a PDF created by LIME. Not only would this be this a nuisance if the piece has something at the very top (or left) of a page, it means that the printed result would be a bit smaller than otherwise (typically 97% for the normal 1/4inch margin). The same would also apply if horizontal centering were not used.

22.2.4. Copy Rectangle deprecated

On Macintoshes, it is strongly recommended that a better way of extracting portions from a Lime file is to create a PDF and use *Preview* (or other PDF facility) to select and copy the required portions. Copy Rectangle is deprecated.

Copying a Rectangle is a kind of printing. On PPC Macs, old versions of Lime created Post-Script. Lime 9.15 does not normally use Post-Script for copying rectangles but uses high (5x) resolution instead. However die-hard PPC users can still create Post-Script by selecting Edit->Copy Rectangle with the Control Key depressed.

On all Macintoshes, Copy Rectangle cannot copy rectangles containing Unicode Text properly. Any text that cannot be rendered in 8 bits in its designated font, will be converted to a question Mark (or equivalent). Furthermore, if the rectangle contains **bold** or *italic* characters in one of the music fonts, other programs, such as *Word* or *NeoOffice*, do not seem capable of using the correct font.

Furthermore, if a copied rectangle is pasted into another application, such as Microsoft's Word, whilst it may look fine on the Macintosh, it often does not if the file is opened on Windows.

So, on Macs, use PDF instead.





22.3. QuickTime and Hearing Pitch Bend and Microtones

QuickTime has a permanent bug (which Apple have said will never be fixed) in that there is a delay between issuing a pitch bend and it taking effect. Lime attempts to work around the problem but, in complex cases, particularly when microtones and pitch bends are used together, the results are by no means perfect, manifested by unexpected jogs at the beginning or end of a sequence.

If you are working with microtones, such as quarter-tones (which need pitch bend to render accurately), it is recommended that the default QuickTime synthesizer is not used. Use either an external Midi device or a 3rd party software synthesizer - *SimpleSynth*³⁶ from <http://notahat.com/simplesynth> is very strongly recommended.

22.4. Long File Names

In 9.15 (and earlier), in the *Batch...* facility on OSX, there is a bug such that, if files with names longer than 32 characters are processed and saved to another directory, the name will be truncated to 32 characters then modified to ensure it is unique³⁷.

Also in the *Batch...* facility on OSX, it is only possible to specify back-up when overwriting a file in place.

When importing MusicXML or NIFF, long file names are also truncated.

22.5. Fonts Dialog

If a font is required that is not one of Lime's standard ones, the *Annotations->Font->font...* menu item invokes a special fonts dialog in which any font can be chosen. This dialog is also invoked to specify a character's font in the Parameters dialog. On Macs, the fonts dialog is normally a composite dialog comprising the standard 'Fonts Panel' coupled with OK and Cancel buttons and a couple of checkboxes to enable specification of the style. Resizing can be done using the upper portion.

Note that, on Macs, the 'Typeface' shown in the fonts panel, identifies a variant of the font and is independent of Lime's bold/italic style. This is why there are separate bold and italic checkboxes.

Because the fonts panel does not work on Mojave (it crashes), a 'simplified' dialog is used. This has a massive pull-down menu of all available fonts, including Lime's standard ones. As on the fonts panel, size and style may be specified.

22.6. Asian and Middle Eastern Annotations

When editing annotations in Asian or Middle Eastern scripts, if the base font for the annotation does not support the necessary glyphs, font fall-back (substitution) is used. In some cases the result when editing may not be the same as when the resultant annotation is formatted. The substitution font for editing may be different and, moreover, moved upwards. After editing, however, the annotation reverts to the correct font and position.

It is recommended that for such scripts, a font designed for the script is used, rather than relying on font fall-back, which may change.

³⁶ In spite of the warning on the web site, it works perfectly well on High Sierra.

³⁷ This is due to the antiquated I/O API that is used, based on the 32 character file limit of OS9.





22.7. Oriental Users

Whilst a few changes have been made to support Unicode file names, by and large Lime's internal structure still uses the old 8-bit MacRoman character set. By patching the Info.plist file inside the Lime execution bundle, it is possible to overcome some of the limitations.

Right-click (ctrl-click) the Lime application; choose 'Show Package Contents'; open the 'Contents' folder; and use a text editor to edit Info.plist. For example patch:

```
<key>CFBundleDevelopmentRegion</key>
<string>English</string>
```

to

```
<key>CFBundleDevelopmentRegion</key>
<string>Japanese</string>
```

22.8. Zoom Limits

On Macs, the maximum zoom level of a page is dictated by the width and height of the whole page. If the page is large, higher zoom levels may not be possible³⁸. The only real impact is on the zoom level of ticker-tape contexts (see section 20) for Lime Lighter. This is because a context is actually 3 nominal pages wide. If the required zoom level cannot be achieved, then it will be necessary to reduce the nominal page width (and hence number of measures) in the nominal context. This is unlikely to be a problem because at high zoom levels, the nominal page width is often much larger than the screen width.

22.9. Music Font Annotations

If you need to enter into an annotation a music font (Marl, Tufa or Sonata) character that is not available using a backslash expression (or if backslash editing is turned off), a 3rd party tool, such as PopChar³⁹, will be needed to enter any character, which does not map to a simple Ascii character (value <= 127). If such a tool is used, always use the Unicode *Private Use Area*. Other areas (apart from simple Ascii) can produce strange results, particularly for Marl.

22.10. Connecting MIDI Devices

On Macs, In order to use an external MIDI device, such as a keyboard or MIDI synthesizer, one needs to use the Mac Utility "Audio MIDI Setup" (in Utilities) to identify the device and the driver to use for it (USB to MIDI Adaptor, probably). Once that is done, Lime should be able to see it and one should be able to select it as the *MIDI Input* or *Output* (under the *Hear* menu).

³⁸ Actual limitation is 4095 pixels in each direction

³⁹ www.ergonis.com/products/popchar





23. Windows Considerations

23.1. Windows Version Support

Lime has been tested on all systems from Windows 2000 up to and including Windows 10.

Note, however, that support for Windows 2000 is deprecated and is unlikely to continue after 9.16; please contact us if this is a problem.

23.2. Refreshing Windows

There is a bug on Windows such that, if another application's window (or a sub-window of the print dialog) is dragged fast across a Lime score window and/or the piano window, it can leave a bit behind or leave small holes in the piano window. Should this happen, the *Refresh* item in the *Window* menu can be used to reconstitute the display.

23.3. Multi-Monitor Environment

23.3.1. Mouse Click on Lime Window Ignored

There is a known issue with Lime on a Windows' set-up with multiple monitors (3 or more, probably). Sometimes mouse clicks on a music window will not work on one of the monitors. If this happens, move the whole Lime window to another one.

23.3.2. Lime Does Not Appear at Start up

If Lime was used on a system with multiple monitors and then subsequently invoked with fewer monitors or with the arrangement changed, it is possible that the remembered position of the main frame window is not on any of the current monitors. If this happens, Lime will start up, but nothing will be visible, except its entry in the task bar.

The solution is to click on the entry in the task bar, then type *Alt* and *Space* together (Alt-Space) to bring up Lime's system menu⁴⁰. Select *Maximize* and Lime should appear on the primary monitor.

⁴⁰ *Alt Space* is a standard Windows short cut for the system menu. 





23.4. Gestures on Windows

On Windows 7 or later, Lime supports gestures on a touch screen device. Unless Lime Lighter is fully active, basic, single finger tapping (and dragging) is the equivalent of using the mouse to click (and drag). This can be used to select notes, bring up menus, select notes, etc. Single finger press and hold is the equivalent of right-clicking; it can be used for annotation selection, etc.

The following specialised gestures are supported:

Two Finger Tap Tap the screen with two fingers simultaneously:
In normal editing mode, brings up a touch friendly Go To BAR dialog that defaults to the first measure of the current page. When Lime Lighter is active, this action is configurable – see section on Lime Lighter.

Press and Tap Pressing one finger onto the screen and then tapping with another maps to the following functions:

<i>direction</i>	<i>short cut</i>	<i>normal function</i>	<i>hear tracking</i>
Left	<i>cmd/ctrl-[</i>	Go to previous page	Slower
Right	<i>cmd/ctrl-]</i>	Go to next page	Faster
Up	<i>cmd/ctrl-shift-[</i>	Go to beginning of piece	Pause/Resume
Down	<i>cmd/ctrl-shift-]</i>	Go to end of piece (last page)	

Note that the alternative “Hear Tracking” is only available when hearing the piece with tracking. If the piece is being heard without tracking, then the normal functions apply.

23.5. Windows Font Smoothing

Lime has a general preference option to control the level of font smoothing, often known as *anti-aliasing*. The options are:

- Off** The Windows’ system option to smooth text fonts on the screen⁴¹ (using *ClearType*) is used. This is **not** recommended because, if system smoothing is turned off then Lime’s music window would be a little rough because not only text annotations but also notes and other music symbol are drawn using text (music symbols use one of the special fonts, Marl, Tufa or Sonata), which are all affected by the font smoothing option.
- Partial** *ClearType* smoothing is enforced for all text and symbols, whether or not it is turned on at the system level. The result is exactly the same as if the Windows’ system option is on.
- On** Additional smoothing, over and above *ClearType*, is used whenever possible (symbols bigger than 17pt on the screen), otherwise *ClearType* is enforced.. This is recommended and is the default.

The preference is for compatibility only, just in case someone finds partial non-anti-aliased fonts easier to see. It is worth noting that, at zoom levels below 2/big (except 1/small), full anti-aliasing is needed for guitar fret grids to display properly on the screen.

⁴¹ On Window 7 this option is in: Control Panel->System->Advanced System Settings->Performance->Visual Effects





23.6. Right to Left and East Asian Languages

Lime supports Hebrew or Arabic (right to left scripts) and East Asian languages, such as Chinese, Korean or Japanese. Of course, Windows must have the necessary material installed. On XP, for example, this is done using the *Regional and Language* control panel; for Hebrew and Arabic, install 'Complex Script and Right to Left Languages'; for Chinese, etc, install *East Asian Languages*.

Currently annotations in such scripts need to be created and edited directly. The annotation helpers will not work.

23.7. PDF Printing

There are issues on Windows with printing large (multi-part, many pages) Lime score at high resolution to PDF files, with anything other than **Adobe PDF** (which is expensive).

Windows 10 has native printing to PDF. Unfortunately Microsoft does not see fit to provide custom scaling. This means that printing to *Microsoft Print to PDF* can only be used if the piece's scaling is 100%⁴², for which it works nicely.

At full resolution, most free Window PDF printers (such as *PDFcreator*, etc) will often, but erratically, lose note-heads on one or two pages of large scores (typically around page 3). Apart from using **lower resolution** (*SHIFT while selecting File->Print, or ctrl+Shift+P*), no work-around has been found and it is believed to be a bug in the printing infrastructure used by these products.

An exception seems to be **doPDF**. It therefore is recommended, that if **Adobe PDF** is not available, **doPDF** should be used for full resolution PDF creation. There is a basic free version and it can be downloaded from:

www.dopdf.com

Note that, when printing a large file to a PDF printer, there can often be a significant delay after the last page has printed while the PDF printing infrastructure processes the result and creates the required file. Patience is needed as there is often no progress indicator (neither **doPDF** nor *Microsoft Print to PDF* provide progress indicators).

So, when printing to PDF on Windows, use low resolution or use **Adobe PDF** or **doPDF**.

23.8. Ctrl+AltGr

Ctrl+Alt+something short cuts must use the left Alt key. The right Alt key (Alt-graphic) will not work for this purpose. This is to avoid confusion with the fact that Windows makes it impossible to distinguish between AltGr⁴³ by itself and with Ctrl.

⁴² A work-around to this absurdity has not yet been identified.

⁴³ Windows actually automatically adds Ctrl to AltGr!





23.9. Bad Midi Devices Crashing PC

Software midi synthesisers and sound cards are handled using system drivers and/or DLLs. If one of these is buggy it can cause Lime to crash or freeze. There is nothing Lime can do to stop it. In order to guard against this happening, whenever a midi device is selected, Lime always sets the preferences to “No Output/Input” before attempting to open the device and only sets it to the device when it has been successfully opened. Thus if Lime crashes or freezes when opening a midi device, it will not happen again when Lime is reloaded. The user will then need to set Midi Output/Input back to a working device.

On some versions of Windows (certainly XP and earlier) some sound cards, notably old members of the “Creative Sound Blaster” family, can cause the whole PC to crash if used as a midi output device. This is because the windows driver advertises itself as midi capable, when it isn’t. The problem only occurs (when it occurs) with old, or emulated devices, which are actually called “*Creative Sound Blaster*”.

To prevent inadvertently choosing a rogue midi device, Lime has an option to detect such devices and blacklist them. By default all devices starting with “Creative Sound Blaster” are blacklisted. You can change this using in the Preferences dialog (edit->preferences). To blacklist one or more devices, ensure the “Blacklist Midi Devices” checkbox is checked and enter their names (case significant) in the blacklisted box. If there is more than one, separate them with a backslash (\). The number of characters up to the backslash are compared, thus the default “Creative Sound Blaster\” will match anything starting with “Creative Sound Blaster”.

There is some evidence that the old *BASSMIDI* causes problems in later versions of Windows; however its successor, *OmniMIDI* seems to work nicely.

23.10. File Name Problems on Windows

23.10.1. Exotic File Names (fixed in 9.16)

In 9.15 and earlier, there was a bug that Windows Lime could not open files with exotic names (e.g. with oriental characters) that cannot be mapped to the standard Windows 8-bit character set. This has now been fixed.

However, you should be aware that, if Lime 9.16 or later is running, but the default Lime (e.g. in *C:/Program Files*) is 9.15 or earlier, opening a file with an exotic name by double clicking on it in the Finder, will not work; it has to be opened from within Lime 9.16. This is because it is first handled behind the scenes by the default version of Lime.

23.10.2. Leading Spaces in File Names

In 9.16 (and earlier), there is a bug in the Windows version which prevents one saving a piece with a file name starting with one or more spaces - the spaces are removed.

23.11. VMware Virtual Machine

If Lime for Windows is run on a VMware virtual machine, the cursor repositioning when exiting low vision scrolling will not work properly, unless VMware’s preferences have “Always Optimize Mouse for Games” set.





24. Lime Lighter

Lime Lighter is a Windows⁴⁴ music stand capability designed for Partially Sighted users. A special licence is needed from Dancing Dots Inc. The main functions of Lime Lighter are:

- Show and highlight music bar by bar or system by system at up to 8x magnification;
- Provide hands-free scrolling using pedals or equivalent;
- Provide automatic scrolling through the music, optionally playing some or all of the parts (particularly useful for practice) and optionally with a metronome;
- Easily navigate to a bar or page;
- Support for touch screen 'gestures' to enable music stand operation without keyboard or mouse.
- Mark up the music using touch screen or mouse;
- Change visual appearance to help each user overcome their particular impairment.

24.1. Manual Scrolling

Low vision tracking (scrolling) is controlled using F13 (go back), F15 (go forwards), and F14 (take repeat or start/stop automatic scrolling) keys. The Lime Lighter pedals map to these keys. The roles of F14 and F15 can be partially swapped using the Lime Lighter Preferences. See section 24.7. The mapping between the actual pedals and the function keys can be changed in the Lime Lighter preferences. In addition, two extra keys, nominally F16 and F17, can be configured when manually scrolling to take an additional disjunction (not repeat) or to turn on the hear option asynchronously. When you do not have or do not want to use a pedal, the function keys can be useful for testing and demonstrating.

- On Windows, *shift-F1*, *shift-F2*, etc, are equivalent to *F13*, *F14*, etc;
- On Macs, *alt-F1*, *alt-F2*, etc, are equivalent to *F13*, *F14*, etc.

Whenever tracking (or low vision scrolling) is initiated, the Piano window is minimized so that it will not obscure the tracked score; it will be automatically restored when tracking is finished. In addition, for low vision scrolling, the music window is made as large as possible and forced to the left of the main screen.

When scrolling is initiated, using the Lime Lighter pedal or key equivalents, the music is first positioned on the measure containing the currently selected note. A second press of the appropriate pedal is then needed to move the rectangle or start automatic scrolling.

24.1.1. TAB and Return Keys

Once Lime Lighter has been activated (with pedal or F13 to F16), the **TAB** key can be used to move forwards or, with shift, backward. Also the **Return** key will go back to the previous repeat or beginning of piece.

⁴⁴ Lime Lighter does, by and large, also work on desktop Macs (without gestures), but not on iPads.





24.1.2. Hearing

If a pedal is configured for hearing, it will invoke the hear option automatically from the current position, optionally with a metronome lead-in (partially controlled by the options for the piece, and partially controlled by the Auto part set options within the Hear dialog). If hearing is already active, the pedal will turn it off. This feature enables a user to practice playing-along and pedalling (or equivalent) with an independent accompaniment. The “*Auto*” part set will be used – see Hear Option.

CTRL+. (control + period) on the computer keyboard will have the same effect.

24.2. Automatic Scrolling

Scrolling can be automatic. In which case, the scrolling is done automatically in accordance with the specified tempos in the music. Optionally the music and/or metronome can sound while automatically scrolling.

Automatic Lime Lighter scrolling is initiated by using F14/F15 or equivalent (middle/right Lime Lighter pedal, depending on the Lime Lighter preferences). It can also be initiated from the Hear menu, but this will not be full screen (see next section).

In automatic scrolling mode, Lime plays the piece from the designated point and moves the low-vision scrolling rectangle in lock-step with the music (aiming to be very slightly ahead). The *Hear->Hear...* dialog is used to specify which parts (from those in the context) will be heard; these parts are saved with the score file. The ‘*auto*’ part set is used for used for automatic low-vision tracking and for the hear option when manually scrolling.

The ‘*Inhibit Auto Hear by Default*’ Lime Lighter preference option controls whether the music/metronome is played while automatically scrolling (see section 24.8). By default, the music and metronome (if configured) are played, but it is possible to configure that nothing is played or just the metronome is played. This can be useful when one just wants to see the scrolling music and, possibly, have metronome clicks (see section 4.2.3), but nothing else.

If you are playing at a point where your part starts immediately (i.e. without a lead-in by another part or if no other parts are being heard), you will almost definitely want a metronome lead-in, otherwise it is almost impossible to synchronise with the scrolled/heard music.

If a pedal is configured for hearing, it can be used during automatic scrolling to turn hearing the music on or off. Note that this does NOT affect the metronome.

24.3. Full Screen Mode

On Windows, if the Lime Lighter zoom level is such that the whole width of the piece is greater than the available width of a maximized window with scroll bars, then the Lime Lighter display will be in a separate view, covering the whole of the screen, including the Windows’ task bar. Apart from the title, showing what piece is active and close/minimize buttons, the rest of the display is dedicated to the score.

Full screen mode is closed whenever scrolling is stopped:

- Explicitly closing the full-screen window;
- Exiting Lime Lighter scrolling;
- Activating any other view.





A full screen window can, however, be moved or collapsed. This can be useful to access Lime's menu. Note that, on Windows 7 or earlier (and on Windows 2000 it's odd), all the restore buttons may not always be visible - over the mouse over the top and the rest should appear.

The full screen window is essentially a dialog associated with the view from which it was invoked (a full screen view is actually a separate sub-view as well). While it is active the original view is inactive and will not respond to any mouse click (except to un-collapse it - when a full screen window is first invoked, the original view is collapsed).

A preference option is available to inhibit use of full screen or to increase zoom automatically to force full screen to be used always. Inhibiting full screen may be needed on old small systems (particularly Windows XP), if there are memory constraints.

24.4. Mark-Up

Within Lime Lighter, either a single finger or a two-finger tap gesture (corresponding to mouse click or shift-click) can be configured to stop Lime Lighter mode and turn on mark-up. If, say, one finger tap is configured to do this, all you need to do to make a mark-up while pedalling is to tap the screen, make the mark-up, then press the pedal to continue.

Mark-ups cannot be done when in Lime Lighter pedal mode, without temporarily stopping it. This is because of the feature interaction between using a touch screen to mark-up, and the gestures for control. Mark-up is turned off automatically when entering any Lime Lighter pedal mode to avoid any confusion. Unless configured otherwise (see below) mark-up mode remains off when Lime Lighter pedal mode is exited.

24.4.1. Printing Marked Up Pieces

By default, when a context is printed, mark ups are included. However there is an option in the print dialog to turn this off and omit mark ups. Optionally the default for printing mark ups can be saved for each context printed.

Note that, while printing a marked up context, mark ups will appear on and jump about the music window. However they are properly restored when printing is complete.

24.5. Ticker-Tape and Autocue Contexts

Ticker-Tape and **Autocue** are viewing modes originally designed for the Lime-Lighter electronic music stand. In Lime Lighter tracking mode, a ticker-tape notation context appears to continually scroll measure by measure like a 'ticker-tape', with no visible system or page breaks, and an Autocue one scrolls vertically system by system. The purpose is to enable a Lime Lighter user to always be able to see the next measure(s) even if they would be on another system or page in a normal page-based score.

Most Lime Lighter users will find that a ticker-tape or autocue context work best for them, see section 20. The one to use will depend on the preferred tracking. Ticker-tape is designed for measure based tracking and Autocue is designed for system based tracking. When tracking in Lime Lighter, a ticker-tape context will always track by measure (bar), regardless of preferences. Similarly, an autocue context will always track system by system.





24.5.1. Mark-Ups

Lime Lighter Mark-Ups are context and page specific. In ticker-tape or autocue mode, mark-ups are **currently only shown when their nominal page is displayed**⁴⁵. If you want a mark-up to be visible at the beginning of one system page when nearing the end of the previous one or not to disappear when leaving a system page, it may be necessary for the Mark-Up to be drawn twice, once on each nominal page.

24.5.2. Switching Between Normal and Ticker-tape or Autocue

The short cut Ctrl/Cmd-semicolon (;) is a convenient quick way of switching between contexts; it goes to the previously viewed context. This can be useful when you sometimes want to view a piece in normal mode (e.g. to discuss it), then start playing it using Lime Lighter. Ctrl/Cmd-shift-semicolon will go to the score context or, if already there, to the last context in the list.

24.6. Lime Lighter Specific Short Cuts

The following short cuts are specific to Lime Lighter:

- | | |
|------------|---|
| cmd/ctrl-` | On Windows (ctrl+reverse quote), with Lime Lighter, set mark-up Draw mode or, with the <i>shift</i> key, turn off draw mode |
| cmd/ctrl-\ | On Windows (ctrl + backslash), with Lime Lighter, turn on, or with the <i>shift</i> key off, the mode to enable the arrow keys (if neither ctrl or alt is pressed) to be used as well as F13-F16 for Lime Lighter navigation. This allows pedals such as <i>AirTurn</i> [™] to be used. |
- Note that *ctrl+alt+* an arrow key will always perform the basic arrow key function, regardless of the Lime Lighter arrow keys setting.

⁴⁵ It is hoped that this may be fixed in 9.17.





24.7. Lime Lighter Clicking & Gestures in Music Window

When Lime Lighter music stand is active, whether with manual or with automatic tracking, mouse clicks, or touch screen taps on the Lime Lighter music window have the following effect:

Right Click	<u>Tap and Hold on a touch screen:</u> Always exits Lime Lighter music stand and returns to normal editing mode. It is the equivalent of the <i>escape</i> key and is not configurable.
Shift-Left-Click ⁴⁶	<u>Tap with two fingers simultaneously:</u> By default will bring up a big, touch screen friendly 'Go To' dialog (for bars - see section 21.17), but may be configured to stop/markup (if needed for short term compatibility) or to do nothing. Note that two finger tap brings up an enlarged Go To dialog (see section 21.17) in normal editing mode, so it is recommended to keep it the same for Lime Lighter.
Left Click	<u>Tap with one finger:</u> By default will be ignored to avoid any possible confusion with other gestures and because some touch screens are over sensitive. It may be configured to stop/markup or to bring up the 'Go To' dialog. Note that, if left click is configured to do anything, double click will be disabled.
Double Click	<u>Tap twice quickly with single finger:</u> Only works if left click is ignored. Otherwise, always brings up the "Go To" dialog for rapid navigation to a bar number (or, in future, rehearsal mark, which may be the default if two fingers are used). This is not configurable.

24.7.1. Configuration of 1 & 2 finger Tap Gestures

The function of one and two finger taps, when in Lime Lighter scrolling mode, can be configured in the Lime Lighter preferences. The options are:

- Do nothing. This may be useful for one finger tap, if your touch screen is over sensitive or if you tend to touch the screen often while playing.
- Navigation Brings up a large, touch screen friendly "Go To BAR" dialog. If user changes bar, Lime Lighter moves to the new designated bar. This dialog allows specification of system page, rather than bar number and relative numbers to be specified by explicitly prefixing the number with + or -. There is also a **Stop** option, to exit Lime Lighter scrolling (equivalent to the *escape* key on a keyboard).

If user was in automatic scrolling, scrolling and play-back will be paused and a press of the middle Lime Lighter pedal will resume it at the new measure.
- Stop/Markup Stops scrolling and turns Mark-Up on. A pedal press is needed to resume scrolling.

High priority for 9.17 is to add a "Go To Rehearsal Mark" capability.

⁴⁶ N.b. shift-left-click is only equivalent to 2-finger tap when the music stand is active.





24.7.2. Press and Tap Gestures

In Lime Lighter mode, the Press and Tap gestures (press one finger onto the screen and then tap with another) can be used instead of the pedals (if you have a hand free) as follows:

Direction	<i>Lime Lighter Scrolling mode</i>	Action
Right	<i>manual:</i>	Advance;
	<i>automatic:</i>	Faster.
Left	<i>manual:</i>	Go back one;
	<i>automatic:</i>	Slower.
Up	<i>manual:</i>	Take Repeat (or disjunction);
	<i>automatic:</i>	Pause/resume.
Down	<i>manual:</i>	Take secondary disjunction (the 'Go To' function');
	<i>automatic:</i>	Pause/resume.

24.8. Lime Lighter Preferences

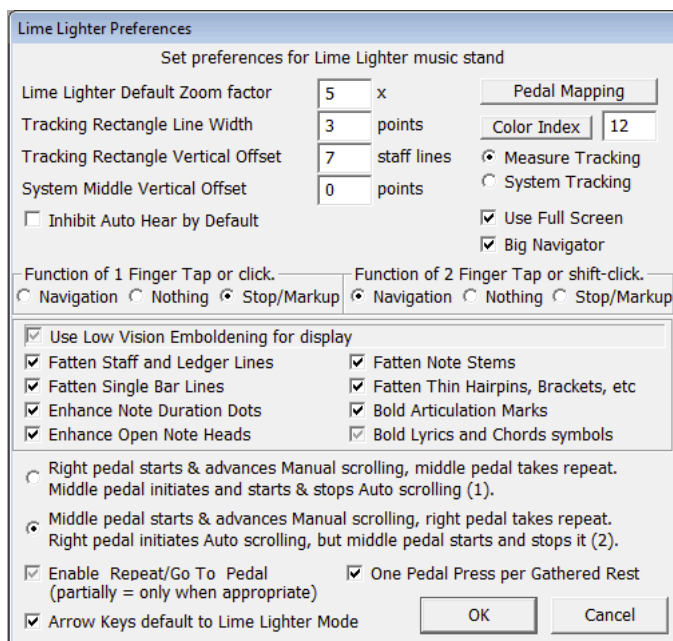
There is a special set of preferences for Lime Lighter. These enable configuration of:

- Zoom to use when Lime Lighting a piece. If this is specified (non-zero), whenever low vision scrolling is invoked, the zoom will change to this value and revert afterwards. + or - may be used for the zoom factor to specify screen/frame width.
Leave this blank (or set it to zero), to use the zoom that is in effect when editing/browsing. Once you have determined your preferred zoom factor, you will probably want to set the Lime Lighting zoom factor to that value.
Note that when using *autocue* system tracking, Lime will force the maximum zoom level to the screen/frame width. With *ticker-tape*, the maximum zoom level is 10.
- Appearance of the tracking rectangle used when scrolling. Note that this is the same as the tracking line used in normal hear tracking.
- Option to specify the offset from the vertical middle of the screen where Lime Lighter positions the highlighted music. The default is in the middle. The value is in actual points on the screen. Note that there are 72 points to an inch (2.5cm).
- Option to partially swap the roles of the middle and right pedals. It is possible to specify that the middle pedal is used for normal manual advancing; this is recommended for new users.
- Option to specify whether full screen mode is used. Unchecked means it is never used; partially checked (default) means it is used if zoom is large enough; fully checked means full screen is always used, increasing zoom to fit if necessary.
- Options to specify the function performed by a 1-finger Tap (simple click) or 2-Finger Tap (shift-click) in the music window when Lime Lighter is active. Note that, on Windows, if a 1 finger tap (mouse click) is configured to do anything, the double click/tap function of bringing up the navigation dialog is disabled.
- Option to specify whether touch screen friendly "Go To BAR" dialog is large or normal sized.





- The 'pedal mapping' button provides a dialog allowing one to specify the function of each of 4 possible pedals. The default number of pedals is 3, but *AirTurn™* devices support up to 4. In the pedal mapping dialog, if you have an AirTurn pedal, the pedal numbers (1-4) refer to the ports on the AirTurn device.
- Option to disable the 'Jump' pedal (taking repeat or Go To) when in manual tracking. If there is no actual repeat or disjunction in the performer's score it prevents inadvertently going back to the start. If partially checked, repeats can only be taken if the currently highlighted measure(s) have a bar with left dots or ones with an explicit Go To or 1st/2nd ending associated with the bar line. If scrolling by system, if there is both a repeat and a Go To, the repeat takes precedence.
Note that, when manually scrolling, if fully enabled, the repeat pedal will go back to the previous right-facing repeat bar, or the beginning of the piece, but if, and only if, the previous pedal operation was a single reverse (usually left pedal).
- Option to move over a multi-measure rest with a single pedal press (default).
- Option to default the use of the up/down/left/right arrow keys for Lime Lighter control. This enables *AirTurn™* pedals (which use Up-, Left-, Down- and Right-Arrows rather than F13 to F16) to be used without modification. If used with either *Ctrl* or *Alt* the arrow keys always have their normal meaning. When used with both *Ctrl* and *Alt*, the arrow keys will always perform their basic function, regardless of this setting.
- When automatically scrolling, Lime can play the music for the parts configured in the *Hear Auto* option of the current context (optionally, with metronome). This can be turned off using the *Inhibit Auto Hear by Default* piece option. If it's checked, there will be no sound; if it is partially checked only the configured metronome will sound. The Go To Bar dialog has an option to temporarily change this.
- Finally there is the emboldening option to turn on a set of parameters to assist in the visibility of the music. If unchecked, the parameters are not applied; if partially checked (recommended) they apply to the screen only; if fully checked, the emboldening parameters are used when printing as well. Within this option, Lime Lighter users can control individual appearance parameters; by default all are set on. If lyric emboldening is partially checked, lyrics and chord symbols are slightly enlarged; if fully checked, they are emboldened as well.



Lime Lighter Preferences





24.8.1. Pedal Mapping

From Lime Lighter's perspective there are virtual 'left', 'middle', 'right' and currently 5 functions controllable by pedals.

Left Go back a measure or system, or start manual scrolling at the beginning of the piece.

Middle The function of the 'middle' and

Right 'right' pedals are governed by a pair of radio buttons in the Lime Lighter preferences:

- 1• The 'middle' pedal starts and advances manual scrolling; whilst scrolling, the 'right' pedal can be used to take a repeat, if so configured (if it is grey, it will only take a repeat if there is one in the current system). If Lime Lighter mode is started by pressing the 'right' pedal, it goes into automatic mode (with sound); thereafter you press the 'middle' pedal to start and stop it and the 'left' or 'right' pedal to go faster or slower. *This is the recommended default for new users.*
- 2• The 'right' pedal starts and advances manual scrolling; whilst scrolling, the middle pedal can be used to take a repeat, if so configured. If Lime Lighter mode is started by pressing the middle pedal, it goes into automatic mode (with sound). *This alternative is available for compatibility with earlier versions of Lime Lighter.*

Go To When manually scrolling, go to a disjunction in the music such as 2nd ending, or explicit 'go to bar', but NOT a repeat. Furthermore if the next bar line is concatenated, and the second line specifies a disjunction, that will be taken in preference to the first (if the 2nd line has no disjunction, but the 1st does, the 1st will be taken).

If you start Lime Lighter with a pedal configured as *Go To*, it will invoke manual scrolling and you will be prompted to enter a bar number to start at.

Hear when manually scrolling, toggles asynchronous play back.

If you start Lime Lighter with a pedal configured as *Hear*, it will invoke automatic scrolling and you will be prompted to enter a bar number to start at.

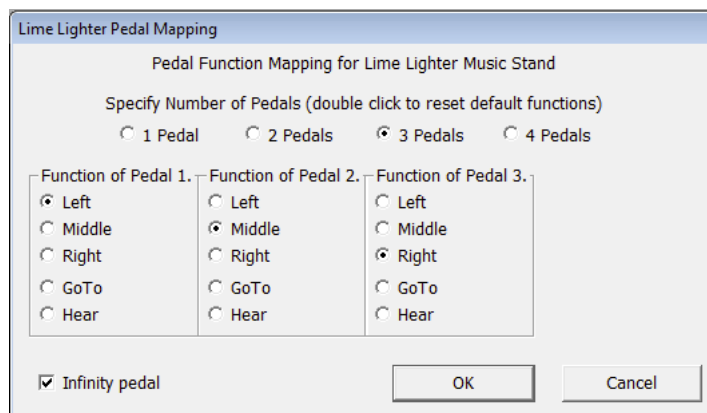
On top of this, you can map the physical pedals on the AirTurn pedal block to the functionality of 'left', 'middle' and 'right'. These are the port numbers on the "digit" box and are numbered 1 to 4 (if you have a 4 pedal device) from the left (note that a sustain pedal would be plumbed into one of the ports). So if you have 4 pedals and, for example, want 1 to be left, 4 to be middle (i.e. manual scrolling) and the other 2 to do absolutely nothing, you should:

- map 1 to 'left'
- map 4 to 'middle'
- map 2 and 3 to right, and, furthermore, disable the repeat pedal in the main Lime Lighter preferences. However it would probably be better to have 1 & 2 mapping to 'left' and 3 & 4 mapping to 'middle'

On a 4 pedal AirTurn device, the buttons on top of the "digit box" have the same function as the pedals (and the corresponding arrow keys).

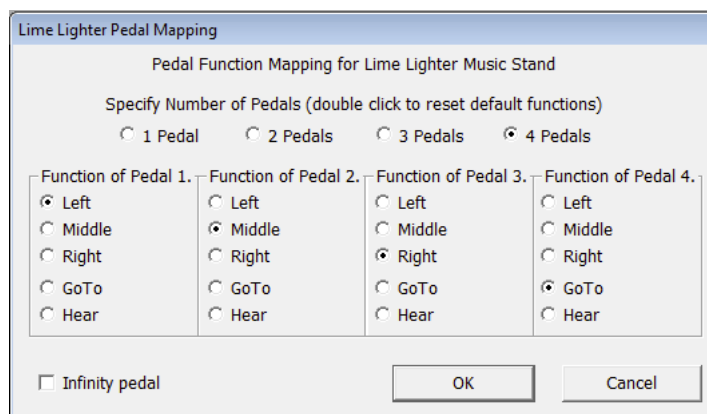
up arrow	= 1
left arrow	= 2
down arrow	= 3
right arrow	= 4



Pedal Mapping - Default for 3 Pedals

It is very important to align the pedal mapping to your actual pedal configuration and with the use of the middle and right pedals in the Lime Lighter preferences.



Pedal Mapping - Possible Configuration for Enabling Secondary 'Go To'

24.8.1.1. Old Infinity Pedal

The *Infinity™* foot pedal, that used to be supplied with Lime Lighter, requires a DLL (*PIEHid.dll*) to drive it. This can be installed in the same folder as Lime.exe. It is recommended that the *Infinity pedal* check box is checked if you are using the old pedal. This is not essential, but will enable a warning if the DLL cannot be loaded.

24.8.1.2. Alternative Pedals

The new AirTurn™ unit can support any switch that operates by making an electrical contact. Pedals are just one example. As well as AirTurn pedals, 3rd party pedals or switches may be used, such as:

- Independent 'Sustain' pedal. Many users, particularly keyboard players, find these more natural.
- Bite switches or equivalent. Whilst not very useful for singers or wind players, some users, particularly those who need their feet, might find them more useful for the basic scrolling operations.
- An option for an organist, for example, may be a bite switch for the basic advancing function (middle), plus two hand operated switches for repeats (right) and back (left).





24.8.2. Lime Lighter Parameter Defaults

Using parameters, Lime has a wide range of options for modifying the appearance of a piece. Many of these can be very useful to improve the visibility of the music for partially sighted users. The “*Use Low Vision Emboldening*” option sets some of these automatically; it is recommended for most Lime Lighter users. The individual check boxes can be used to enable or disable specific changes.

Note that these are the defaults used on your computer⁴⁷; they are not saved with the piece; they are the default parameter settings and may be overridden by any parameters in a piece. If the option is partially checked (default) the parameter defaults apply to the display only, if fully checked they will also be used for printing pieces.

24.8.3. Lime Lighter Arrow Keys

The old *Infinity* pedal unit can be used directly as the pedal uses F13, F14 and F15. However the new *AirTurn*™ pedals use Up-, Left-, Down- and Right-Arrows rather (than F13 to F16). This conflicts with the standard use of these arrow keys. A preference option, *Edit->Preferences->Lime Arrow Keys* allows one to temporarily specify that the Up-, Left-, Down- and Right-Arrows should be used for Lime Lighter scrolling with the *AirTurn*™ pedals or for their normal function⁴⁸.

The short-cut CTRL+\ (control + backslash) can be used to quickly turn on Lime Lighter mode and CTRL+SHIFT+\ (control + shift +backslash) can be used to turn it off. If one of these short cuts is used and the arrow keys are already in the required mode, nothing will be done (except a double beep), otherwise the user must confirm the change. On Windows, CTRL+ALT+\ (control + alt +backslash) will toggle the arrow key mode (like the corresponding menu item), which will always require confirmation.

If used with CTRL and ALT, Up-, Left-, Down- and Right-Arrows always perform their basic function, regardless of whether they are set in Lime Lighter mode.

24.9. Lime Lighter on Macs

Lime Lighter is experimental on Macintosh desktop computers (it unlikely to ever work on iPads and *Lime itself will NOT work on MacOS 10.15 (Catalina) or later*)

Main issues are:

- Zoom level limitations for wide ticker-tape contexts (see section 22.7). To achieve the required zoom, it will probably be necessary to reduce the nominal width of a page and associated number of measures in the context.
- Measure numbers are not erased [yet] when the highlighting rectangle is moved.
- Touch screen facilities are not available as no desktop Mac supports them.
- Only the new *AirTurn*™ pedals are supported. Note that, on Macs, Alt-F1 to Alt-F4 are equivalent to F13 to F16.

⁴⁷ The use of modified parameter defaults is much more efficient than the use of Parameters per se, which particularly for note heads, do reduce Lime’s performance and can impact on responsiveness to pedal presses.

⁴⁸ Note that there is a Lime Lighter preference that enables one to default the arrow keys to Lime Lighter mode.





25. Lime Aloud

Lime Aloud is a special version of Lime for Windows, which provides a wide range of verbal and musical cues to make it relatively easy for the blind or partially sighted to use Lime independently, either for learning scores or composing/notating music, or both. Though it normally uses JAWS, basic facilities are available with other screen readers or even with no screen reader at all.

25.1. Lime Aloud - Special Speaking Options

The Lime Aloud user can control whether Lime speaks details of selected music or annotations, and to what level of detail. The level of detail provided by Lime when notes are clicked or arrow-keys are used to navigate, can be controlled using preferences. This affects which classes of annotation are described (spoken). Currently three levels of detail are possible in addition to the basic information about the selected note, such as position (bar, beat, etc) and the note details (notated pitch, articulation, ornaments, etc):

1. Directions. All annotations that provide musical direction, including dynamics, tempo, pedals, pauses, mood, etc. Hairpins are included at this level.
2. Chords/lyrics/fingers and rehearsal marks. In addition to directions, any chords, lyrics, fingering or rehearsal mark annotations are also spoken. *This is the default for new users.*
3. Environment/lines. In addition to the first two levels, all other visible annotations (such as titles, copyright, etc), including lines and curves will be described.

As well as enabling speaking (enabled by default for Lime Aloud), preferences allow one to specify that the spoken messages are logged to the console. Though primarily for error reporting & debugging, this may be useful for some.

25.2. Toggling Lime Speaking

At any time when the music or piano window is active, the Lime automatic speaking option can be temporarily toggled on or off by pressing the Shift and Escape keys together (shift-Escape)

Note that shift-Escape only affects the commentary explicitly provided by Lime; it does not affect menu and dialog commentary, provided by screen reading applications, such as JAWS on Windows or VoiceOver on Macs. Furthermore, it only affects commentary when using the mouse or arrow keys for navigation; explicit requests for commentary and piano-window short-cut feed-back are unaffected.

25.3. Hearing Status

In normal editing mode, the bottom row of alphabetic keys (Z-M), will cause status information to be spoken; e.g. M provides the piece file name, and B the bar number. Normally the N key gives the status of note entry, if active. However, when hearing is active, because note entry is not possible, the N key will give the hearing status. While hearing is active, the other keys behave as normal, e.g. B will tell you the current bar number.

25.4. Progress Messages

When performing a time consuming operation, Lime puts a status update in the music window header. In some cases, this includes a rotating bar. These messages will be spoken periodically by lime Aloud. The rotating bar, indicated by speaking "*tik*" or "*tok*", implies that Lime is actively progressing.





25.5. Commas and Periods

Lime aloud assumes that commas and periods will cause pauses but not be spoken explicitly.

25.6. Navigation Short-Cuts

Currently a number pad (turned off) is fairly essential in order for unsighted users to use the wide range of number-pad short cuts that Lime provides, see section 5.4.

25.7. JAWS Issues

Lime Aloud for JAWS includes a JAWS' script to be used for Lime, which must be installed. At least version 9.16.4 is needed for Lime Aloud dialogs to work properly.

25.7.1. Dialogs

With the aid of Lime's JAWS' script, Lime tries to make its dialogs as accessible as possible. The following are worth noting:

- For the Lime application, Lime's JAWS' script forces two settings for all verbosity levels associated with speaking about dialogs:
 - "*Speak Dialog Text*" is always turned OFF. This is so that Lime (and the scripts) are better able to control what is spoken when.
 - "*Speak Control Group Name*" is always turned ON. This is because Lime often uses it (sometimes invisibly) as a prefix to identify controls within the group.
- When Lime's JAWS script is active, many dialogs offer some help (JawsKey is usually the *Insert* key, but may be configured in JAWS to something else):

JawsKey + Slash	offers basic help on the dialog if available
JawsKey + Shift+Slash	speaks more advanced help on the whole dialog, if available (JawsKey + question mark)
JawsKey + Tab	speaks the currently focussed control within a dialog (JawsKey + Alt+Slash currently does the same thing).

25.8. Alternatives to JAWS

Lime Aloud is primarily designed to work with JAWS. However many aspects will work with other screen readers and even with no screen reader at all. If JAWS is not available, or you want to experiment with another screen reader, there is a Lime Aloud preference option to specify that JAWS should not be used

: *Use JAWS, if installed.*

If this is not checked, the built-in Windows' text to speech facilities (SAPI) will be used. Furthermore, if this is not checked, there is an option

: *Speak additional dialog information.*

Which, if checked, will cause Lime to speak basic information about dialog when they are invoked; if fully checked, this will include the dialog name. If you change whether JAWS is used or not, you will be asked to confirm the change and whether the change should be temporary for the current Lime session (default) or permanently saved in the preferences.

Clearly if JAWS is not being used by Lime, it is best if JAWS is not running. If JAWS is active and Lime is not using JAWS, stuff can be spoken simultaneously, particularly if "*Speak additional dialog information*" is checked.





26. Lime-GoodFeel Brailleing

If authorized, Lime-GoodFeel together with JAWS, can provide real-time brailleing of Lime music on a braille terminal.

26.1. Lime-GoodFeel Preferences

There are now a couple of Lime specific preferences that can be set by the user using the new *Edit->Preferences->GoodFeel Preferences* dialog:

- Enable Real Time Brailleing.** If this is unchecked, no brailleing at all will be done by Lime. Normally this should be checked. If you uncheck this, you will be asked whether you really want to turn off Lime's brailleing.
- Braille Line Width.** This specifies the line width of your braille terminal. If zero (or empty, the default) the display width specified in GoodFeel's preferences will be used (default 32). It is recommended that this is normally left blank and the width is set using GoodFeel itself (4.0.1 or later is needed).
- Note that, under some circumstances (when using an old version of the DLL) Lime's brailleing does not split the braille into lines properly. If that's the case, setting it here to the width of your device is essential.

If the Lime GoodFeel preferences are opened with the shift key down, there are two additional check boxes:

- Log this Session,** turns on debug logging of brailleing.
- Use GoodFeel Program,** identifies whether to use the old technique of invoking the GoodFeel program itself to do the real-time brailleing on behalf of Lime. Nowadays, Lime does it itself, much more efficiently, using a special GOODFEEL.dll (if available), but will revert to the old technique of there are problems with the DLL. The primary use of this checkbox is to see what Lime is actually using.

26.2. Braille Window

When brailleing is active, as well as braille on the braille terminal, a copy of the braille is shown in a separate conceptual braille window (sighted users see it as braille, unless a different font has been chosen).

Lime's Window menu provides options to *Show* and *Hide* the Braille window, in which a copy of the Braille is displayed. When this window is focused, the user can navigate through the 'written Braille' and the selected note in the Lime music window will change to reflect the position in the Braille window.

When the focus is on the Braille window, its system menu can be used to focus back to Lime. The shortcut for this is **Alt+space L**.

When the focus is on the Lime music window, the short cuts are:

- Alt+W B** to show the Braille window and focus on it
- Alt+W H** to hide the Braille window.

If the Braille window is actually closed, the Lime Aloud user will be informed. Showing it will cause it to be re-opened, but focus will remain on the Lime music window.





26.2.1. Braille Font

Normally the Braille window uses a braille font, so it looks like it would feel. For sighted debugging purposes, it is possible to set another font for the window, using the font option in the braille system menu (*Alt+space+F*). If this is invoked with shift, it will default to the braille font, otherwise it will default to whatever the current font is.

Note that the font always reverts to Braille, when Lime is reloaded.

26.3. Brailleing Timeout

Real-time brailleing of Lime scores is done by Lime using a combination of GoodFeel and JAWS. In the event of GoodFeel not being able to provide the necessary braille, Lime will time-out waiting for a response and report an error that "*GoodFeel seems to have hung*" and asks whether the user wants to go on waiting. The user has the options:

Yes (alt+Y) to wait for another period.

No (alt+N) to stop waiting for the moment.

Cancel (alt+C)- to stop brailleing the current view of the piece for this session.

If Cancel is used, no more braille will be produced for the current view. If, however, a new view is created, then brailleing will be re-attempted, in the hope that the GoodFeel problem has been resolved.

26.4. Archaic GoodFeel 3.2

If you are still using GoodFeel 3.2, contact Dancing Dots to get an upgrade.

In the mean time, in order for GoodFeel 3.2 to accept the newer file format, you may need to edit the *goodfeel.ini* file in the c:\windows folder (or wherever it is located). Find the line that says:

```
allow_newer_file_format=0
```

and change it to

```
allow_newer_file_format=1
```

GoodFeel 3.2 does not work with Lime 9.16 or later - so get an upgrade to 3.5 or later now (4.0 or later is strongly recommended)!

