Lime 9.19 Documentation Addendum

Annex to the Lime™ Manual
for version 9.19.6.
(work in progress)
2026-10-06

Table of Contents

1. Introduction 13
1.1. Summary 13
1.2. System Requirements 14
1.2.1. Windows 14
1.2.2. Apple Macintosh 14
1.2.2.1. Catalina Warning for MAC Users 14
1.3. Web Site 14
2. Registration 14
3. Annotations 15
3.1. Annotation Selection – Right-Click 15
3.2. Unicode Support 15
3.2.1. Code Pages (and conversion to Unicode) 15
3.2.2. Pre-Unicode Fonts on Old Macs 16
3.2.3. Unicode Fonts 17
3.3. Annotation Anchoring 17
3.3.1. Horizontal Anchoring 17
3.3.2. Vertical Anchoring 18
3.3.2.1. Chord Ornament-Fixed Anchoring 19
3.3.2.2. System Anchoring 19
3.3.2.3. Staff Spacing Scalable Style 19
3.3.2.4. Line/Curve Vertical Growing 20
3.3.2.5. Extended Line/Curve 20
3.3.3. Text Justification 20
3.3.3.1. Horizontal Locking 21
3.3.4. Leave Horizontal Space 21
3.3.5. Text Annotation Placement Dialog 21
3.3.6. Line/Curve Placement Dialog 22
3.3.6.1. Curve Dimensions 23
3.4. Text Editing 24
3.4.1. Editing Existing Annotations 24
3.4.2. Pasting Text into Annotations 24
3.5. Constrained Drawing and Moving 24
3.5.1. Shift when Creating a Line or Curve 24
3.5.2. Shift when Modifying the Size of a Line or Curve (or Shape) 24
3.5.3. Shift when Dragging an Annotation 25
3.5.4. Alt when Dragging an Annotation (high resolution, slow dragging) 25
3.6. Fonts 25
3.6.1. Replace one Font by Another 25
3.6.2. Font Names 26
3.7. Backslash Codes in Annotations 26
3.7.1. New Backslash Codes 26
3.7.2. Backslash Editing Option 27
3.7.3. Backslash Editing Mode 27
3.8. Stack Vertically and Fingering 28
3.9. Fingering and String 28
3.10. Auto-generated Text Annotations 30
3.11. Multi-Line Text 31
3.12. Undo and Redo 31
3.13. Extending Underlines and Hyphens in Lyrics 32

3.13.1. Terminating Underline Extension 32
3.14. Staff Size Adjusting 32
3.15. Annotation Assistants 33
3.15.1. Chord Symbol and Fred Grid Assistants 33
3.15.1.1. Old Chord Symbol Sizing 33
3.15.1.2. Chord Symbol Naturals 33
3.15.2. Line Bracket Assistant 34
3.15.3. Other Line/Curve Assistants 34
3.16. Annotation->Text (Category) Menus 34
3.16.1. Text with Marl+defaults 34
3.16.2. Annotation Classifications 35
3.16.2.1. Figured Bass 35
3.16.2.2. Poet and Dedication Annotation Classification 35
3.16.3. Playback Interpretation 35
3.17. Lines and Curves 36
3.17.1. Line Types 36
3.17.1.1. Rectangles 36
3.17.1.2. Arrowed Lines 36
3.17.1.3. Dashed Lines or Curves 36
3.17.1.4. Zigzag Lines 37
3.17.2. Curve Types 37
3.17.2.1. Ellipses and Circles 37
3.17.2.2. Phrase Indicator 37
3.17.3. Line Clipping 38
3.17.4. Line and Curve Flipping 38
3.17.5. Curve Inversion 38
3.17.6. Curve Line Width 38
3.17.7. Line Staff Size Adjusted 38
3.17.8. Resizable (Extended) Lines and Curves 38
3.17.8.1. Resizable Line and Curve Selection 39
3.17.8.2. Clearing Extendibility 39
3.17.8.3. Anchoring Extendable Lines and Curves 40
3.17.8.4. Extended Curves 40
3.18. Partially Visible Annotations 40
3.19. Duplicate Annotations *at Every System / on Every Page / in Every Part Context* 41
3.19.1. Duplicate in Every Part Context 41
3.19.1.1. Invisible Annotations that have been Duplicated 41
3.19.2. Deleting Annotations that have been Duplicated to Every Something 41
3.19.3. Modifying Text Annotations that have been Duplicated 42
3.19.4. Redundant Duplication 42
3.20. 1st Anno, on or after Note 42
3.21. Group Selection 42
3.22. Shoving Annotations 42
3.23. Allow Any Graphic Alt-Right-Click 43
3.24. Annotation Selection by Class 44
3.24.1. Select All Annotations 45
3.25. Remember Selected Annotations 45
3.26. Copy (duplicate) Annotations *To a Specific Voice* or *a Specific Place* 46
3.26.1. Destination Place 47
3.26.2. Selecting Annotations to Duplicate 47
3.26.3. Examples 48
3.26.3.1. To a Specific Voice 48
3.26.3.2. To Specific Place 48
3.27. Annotation Creation Mode= 49

4. Hearing 50
4.1. Short Cuts 50
4.2. Hear Dialog 50
4.2.1. Parts to Hear 51
4.2.2. Metronome 51
4.2.2.1. Metronome and Time Signature 52
4.2.2.2. Metronome Accuracy 52
4.2.2.3. Metronome Pitch and Channel 52
4.2.3. Tracking 52
4.2.4. Skip Repeats 52
4.2.5. Hear Selection and Looping 53
4.2.5.1. Stop After Bar (measure)... 53
4.2.5.2. Begin At Bar (measure)... 53
4.2.5.3. Looping... 54
4.2.5.4. Start At and Stop After Special Values 54
4.2.6. Do Not Reset Midi 55
4.2.7. Ignore Sustain Ped 55
4.2.8. Use Piano for Voices with No Instrument 55
4.2.9. Speak a Voice's Lyrics or Chord Symbols 55
4.2.10. Louder Parts 56
4.2.11. Voices in Context 56
4.2.12. No Parts Selected 56
4.3. Hear to End of Piece 56
4.4. Default MIDI Instruments (Programs) 57
4.4.1. Piano Sounds 57
4.4.2. Silent Voice 57
4.5. Resetting MIDI 57
4.5.1. Detecting New MIDI Devices 58
4.5.2. No Sound 58
4.6. Playable Pitch Range 58
4.7. Microtones, Pitch Bend and Portamentos 59
4.8. Modify Playback 59
4.9. Re-compute Playback 60
4.10. Basic Drum Mapping 60
5. Short Cuts 61
5.1. Menu Short-Cuts 61
5.2. Changed Default Short Cuts 61
5.3. User Defined Short Cuts 62
5.4. TAB, Arrow Keys and Number Pad Arrow Keys Short Cuts 62
5.5. Playing (and Summarizing) Selected Note or Chord 63
6. Reading Older Scores 64
6.1. Upgrading Issues with Pre 9.15 Scores 64
6.1.1. Annotations on Invisible Rests 64
6.1.2. Invisible Rests become Visible 64
6.1.3. Stem Directions 64
6.1.4. Annotation Horizontal Placement 65
6.1.5. Page Width 65
6.1.6. Tuplet Plotting 65
6.1.7. Mark and Accent Placement 65
6.1.8. Old Scores between Macs and Windows 65
6.1.9. Fonts on Macs 65
6.2. 9.17 Format 66

6.2.1. 9.17 and Later Features ... 66
6.2.2. Reading Older Files ... 66
7. Import and Export ... 67
7.1. MusicXML Importing ... 67
7.1.1. MusicXML File Name Extensions ... 67
7.1.2. Unicode Support ... 67
7.1.3. Creator Funnies ... 67
7.1.4. Timing Errors and Divisional Faults ... 68
7.1.5. Notation Ornaments ... 68
7.1.6. Fingering and String Specification ... 68
7.1.7. Dynamics and Micro-Tones ... 68
7.1.8. Tuplets ... 68
7.1.9. Slurring ... 69
7.1.10. Tempo Marking ... 69
7.1.11. Importing Options ... 69
7.1.12. SMuFL (**S**tandard **Mu**sic **F**ont **L**ayout) ... 72
7.2. MusicXML Exporting ... 72
7.3. NIFF Import ... 73
7.3.1. SharpEye ... 73
7.4. NIFF Export ... 73
7.5. Scanned Music ... 73
8. Problem Fixing ... 74
8.1. Duration Errors ... 74
8.2. Fixer Contexts ... 74
8.2.1. Voices in a Fixer Context ... 75
8.2.2. Ticker-Tape ... 75
8.2.3. Oddities ... 76
8.2.4. Saving ... 76
8.3. Duration Editing ... 76
8.4. Finding Duration Errors ... 77
8.5. Pitch Violation ... 77
8.6. Deleting Idle, Non-Printing Voices ... 77
8.7. Fix Missing Reminders ... 77
8.8. Clear Spurious Context Data… ... 78
8.9. Fix End of System Bar Lines… (and bar Numbering Inconsistencies) ... 78
8.10. Check Page Count… ... 78
8.11. Corrupt or Faulty Lime File ... 78
9. Colours ... 79
9.1. Music Window Appearance ... 79
9.2. Colour Palettes ... 79
9.2.1. Customisation ... 80
9.2.2. Inverted Colours ... 80
9.3. Colouring Note Heads and Keys by Note Name ... 80
9.3.1. Default Note Name Colour Mapping ... 81
9.4. Colouring Issues ... 81
10. System and Staff Distance Layout Options ... 82
10.1. Absolute Staff/System Separation Parameters ... 82
10.2. Additional Before Staff/System Separation Parameters ... 82
10.3. Staff Drag ... 83

11. Parameters ... 84
11.1. Default Values ... 84
11.2. Number Values ... 84
11.3. Character Values ... 84
11.4. Normal Size and Cue/Grace ... 84
11.5. Note Head Widths ... 85
11.6. Line Widths ... 85
11.7. Parameter Notes ... 85
11.7.1. Y offset from top staff line to number of bars ... 85
11.7.2. Whole and Half Rest Characters ... 85
11.7.3. Page Number (Footer) Position ... 85
11.7.4. Shared Note Head Discrepancy ... 86
11.7.5. Bar Line Left of Single-Staff System ... 86
11.7.6. Slurs Over System Breaks ... 86
11.7.7. Ties Like Opposite Bowing to Slurs ... 87
11.7.8. Tied Chords Like Only One Tie Curve Drawn ... 87
12. Zoom Layout ... 88
12.1. Default Zoom ... 88
12.2. Temporarily Zoom to Fit Screen ... 88
13. Bar Lines ... 89
13.1. Restart Bar Numbering ... 89
13.2. Hidden Bar Lines ... 89
13.3. Hearing Options ... 90
13.4. Concatenated Bar Lines ... 90
13.5. Contexts ... 90
14. Slurs and Ties ... 91
14.1. Slur Association ... 91
14.1.1. MusicXML Importing Slurs ... 91
14.2. Adding Slurs and Ties ... 91
14.3. Slur End Placement on Stem or Note Head ... 92
14.4. Grace Slur ... 92
14.5. Additional Slur ... 92
14.6. Ties as Slurs ... 93
14.7. Conversion of Old Files ... 93
15. Hidden Objects ... 94
15.1. Hidden Annotations ... 94
15.2. Hidden Symbols ... 94
15.3. Hidden Rests ... 94
15.4. Hidden Key Signatures and Clefs ... 94
15.4.1. Hidden Key Signatures ... 95
15.4.1.1. 8va and 8vb Passages ... 95
15.4.1.2. Default Accidentals ... 95
15.4.1.3. Showing Invisible Key Signatures ... 95
15.4.2. Hidden Clefs ... 95
16. Preferences, Context Options and Piece Options ... 96
16.1. New Preferences ... 96
16.1.1. Make ~Backups when Saving ... 96
16.1.2. New or Old file format Warnings ... 96
16.1.2.1. Setting Default Character Set ... 96

16.1.3. Emboldening Preference ... 96
16.1.4. Metronome is default when Hearing ... 97
16.1.5. Highlight Notes with Unplayable Pitch ... 97
16.1.6. Old Default Accidentals ... 97
16.1.7. Music Window Flashing Rate (of insertion point) ... 98
16.1.8. Reminders to Save File ... 98
16.2. Context Options ... 98
16.2.1. Automated Footer ... 98
16.2.2. Showing Accidentals ... 99
16.2.2.1. Automatic Accidental Showing ... 99
16.2.2.2. Separate Grace Accidental Showing ... 99
16.3. Piece Options ... 99
16.3.1. Uniform Marks as in 9.05 and earlier ... 100
16.3.2. Reminder Clefs and Keys ... 100
17. Polymetric and Polytemporal Parts ... 101
17.1. Polymetric Parts Option ... 102
17.2. Note Editing ... 102
17.3. Beaming ... 102
17.4. Bar Numbering Inconsistencies ... 102
17.5. Polytempo Note Duration Multiplier (very advanced feature) ... 103
17.5.1. Re-computing Playback ... 104
17.5.2. Linear Spacing ... 104
17.6. Systems... ... 104
17.6.1. Bulk Breaks with the Page->Systems... option ... 104
17.6.2. Manual Insertion of System and Page Breaks ... 104
17.6.3. Measure Insertion and Deletion ... 105
17.6.4. Further Study ... 105
18. Parts and Voices ... 106
18.1. Parts and Voices on staff... Dialog ... 106
18.1.1. Arrangement Propagation ... 106
18.1.2. Apply ... 106
18.1.3. Previous and Next ... 106
18.1.4. Don't Print ... 106
18.1.5. Move Staff... ... 106
18.1.6. All Staves ... 107
18.1.7. No Ledger Lines ... 107
18.1.8. Changes to Number of Staves ... 107
18.1.9. Part or Voice Change ... 107
18.1.10. Uniform Stem Directions ... 107
18.1.11. Grace Notes ... 108
18.2. Voices in Parts... ... 108
18.3. Idle Staff Hiding ... 109
19. Quarter Tones and Special Accidentals ... 111
19.1. Quarter Tone Special Accidentals Dialog ... 111
19.2. Note Entry ... 113
19.3. Hearing Special Accidentals ... 113
19.4. Special Accidentals Parameters ... 113
20. Ticker-Tape (panorama) and Autocue Contexts ... 114
20.1. Setting Up a Ticker-Tape (panorama) or Autocue Context ... 114
20.2. Navigation ... 115

20.3. Nominal Page Width of a Ticker-Tape Context 115
20.4. Nominal System Separators in a Ticker-Tape Context 115
20.5. Page Size & Zooming 115
20.6. Editing 116
20.7. Annotation Anchoring 116
20.8. Printing 116
21. Piano Windows 117
21.1. Horizontal Piano Window 117
21.1.1. Piano Window Mode 117
21.1.1.1. Setting Mode Menu Item 118
21.1.2. Piano Window Music Mode Buttons and Status Indicators 118
21.1.3. Annotation Mode Status Indicators 118
21.1.4. NUM-Lock Indicator 118
21.1.5. Piano Window Zoom menu 119
21.2. Vertical Piano Window 119
21.2.1. Vertical Piano Position 120
22. Miscellaneous Items 121
22.1. About Lime 121
22.2. Edit->Clear 121
22.3. Dialogs 122
22.3.1. Entering Numbers in Dialogs 122
22.3.2. Tri-State Check-boxes 122
22.4. Note Entry 122
22.4.1. Grace Notes 122
22.4.1.1. Nachschlag Grace Notes After a Note 122
22.4.2. Default Accidentals 123
22.4.3. Transposing Instruments 123
22.4.4. Replacing Tuplets by Other Tuplets 124
22.5. Key Signature Dialog 124
22.5.1. Short Cut to Set Sharps or Flats 124
22.5.2. Default Accidentals 124
22.5.3. Transposing Instrument 125
22.5.4. Octave Shifted Play-Back 125
22.5.5. Showing Naturals Before Bar Line 125
22.5.6. Applying to Selected Staves 125
22.5.7. Deleting Key Signatures 125
22.6. Clefs Dialog 126
22.6.1. Clefs 8va and 8vb 126
22.6.2. Percussion Clefs – Drum Mapping 126
22.6.2.1. Clefs' Percussion Mapping Dialog 127
22.6.2.2. Percussion Key Signature 127
22.6.2.3. Note Entry 127
22.7. Transposing 128
22.7.1. Transposing Parts 128
22.7.2. Advanced Note Transposing – Tune Inversion 128
22.8. Showing Tuplet Numbers 128
22.9. Accidentals 129
22.9.1. Parenthesized Accidentals 129
22.9.2. Showing Accidentals (by Voice) 129
22.10. Tempo Annotations are Metronome Markings 129
22.11. Rulers 129

22.12. Low Resolution Rendering ... 130
22.12.1. Printing ... 130
22.12.2. Copy Rectangle ... 130
22.13. Time Signatures ... 130
22.13.1. Advanced Time Signatures ... 130
22.13.2. Time Signature Parentheses ... 131
22.13.3. Placement ... 131
22.13.4. Beaming Rule ... 131
22.13.5. Polytemporal ... 131
22.14. Measure Insertion and Deletion ... 132
22.15. Symbol Placement ... 132
22.15.1. Bulk Changes – Systems Option for Symbols at End of System ... 132
22.15.2. Manually Splitting Systems ... 132
22.15.3. Clef, Key Signature and Time Signature Dialogs ... 132
22.16. Next, Previous Buttons and Apply Buttons in Symbol Dialogs ... 133
22.16.1. Apply ... 133
22.16.2. Previous, Next and Reset ... 133
22.17. Saving ... 133
22.17.1. Locking (Publishing) Scores ... 134
22.17.2. Undo After Save ... 134
22.18. Go To Bar and Go To Page ... 134
22.18.1. Go To Bar/Page Dialog ... 134
22.18.1.1. Relative Measure or Page Number ... 135
22.18.1.2. Next or Previous Repeat Bar ... 135
22.18.1.3. Bar Numbering Disjunctions ... 135
22.18.1.4. Former Place ... 135
22.18.2. Going to Beginning of Current Page ... 136
22.18.3. Going to First or Last Page ... 136
22.19. Selecting a Note ... 136
22.19.1. Go Back to last selection ... 136
22.19.2. Note. ... 136
22.19.3. Selecting Invisible Notes or Rests ... 136
22.20. Voice and Part Management ... 136
22.20.1. Merging Staves ... 136
22.20.2. New Part Creation ... 137
22.21. Rhythm Errors and Voice Length Consistency Checks ... 137
22.21.1. Next Rhythm Error ... 137
22.22. Don't Share Note Head ... 137
22.23. Switching Contexts ... 137
22.24. Select All Notes or Annotations ... 138
22.24.1. First Bar and Last Bar Special Values ... 138
22.24.2. Select All Notes ... 139
22.25. Beaming over Rests ... 139
22.26. Multi-Length Staff Chords ... 140
22.26.1. Accumulated Arpeggios ... 141
22.27. Arpeggio Ornament Annotations ... 142
22.28. Braces ... 143
22.29. Bar Numbers at System Start ... 143
22.30. Midi File Export ... 144
22.31. Midi File Import ... 144
22.32. Internal Fields ... 144
22.32.1. Navigation when Inspecting Internal Fields ... 144

22.33. Bar Lines Through First Chord (advanced feature) ... 145
22.34. Align Bars... ... 146
22.35. System Breaks ... 146
22.35.1. Re-Pagination (Auto-Paginate) ... 146
22.35.2. System Breaks from List ... 147
22.35.3. Ignoring Timing Option ... 147
22.35.4. Reminders ... 147
22.35.5. Staff Drag ... 147
22.36. Copying and Pasting Music ... 147
22.36.1. Pasting Music ... 147
22.36.2. Copying and Pasting Measures ... 148
22.36.2.1. Copying Measures Between Contexts ... 148
22.37. Note Stem Flags ... 149
22.38. New Note Heads and Accents ... 149
22.39. Symbols at End of Staff ... 149
22.40. Grace Note Rests ... 149
22.41. Guitar Tablature ... 150
22.42. Default Accidentals... Menu Option ... 151
22.43. Swing Eighth Note Pair ... 151
22.44. Beaming Rule ... 151
22.45. Tremolo Beaming ... 151
22.46. Enabling Combined Rests ... 152
22.47. Multi-Measure Rest Bar Width ... 153
22.48. Inserting Symbols before Combined Rests ... 153
22.49. Batch File Processing ... 154
22.49.1. Destination Specification ... 154
22.50. Lime Console ... 154
22.51. Remember Voice Chord ... 155
22.51.1. Selecting Remembered Note ... 155
22.51.2. Curve or Line Drawing ... 155
22.52. Shift Key Menu Alternatives ... 156
22.53. **Note Ornament to Play** ... 157
22.54. Lime Score Format with Graphics ... 158
23. Max OSX Considerations ... 159
23.1. OSX Version Support ... 159
23.1.1. Catalina, 10.15, **Warning** ... 159
23.1.2. Mojave, 10.14 ... 159
23.1.3. Yosemite, 10.10 ... 159
23.2. Printing ... 160
23.2.1. Post-Script Printing Deprecated ... 160
23.2.2. Print Centering ... 160
23.2.3. Copy Rectangle deprecated ... 160
23.3. QuickTime and Hearing Pitch Bend and Microtones ... 161
23.4. Fonts Dialog ... 161
23.5. Asian and Middle Eastern Annotations ... 161
23.6. Zoom Limits ... 162
23.7. Maximizing ... 162
23.8. Music or Symbol Font Annotations ... 162
23.9. Connecting MIDI Devices ... 162

23.10. Use of WINE ... 163
23.10.1. Installation ... 163
23.10.1.1. C-Drive on Porting Kit or CrossOver ... 163
23.10.1.2. C-Drive on other Wrappers ... 163
23.10.2. Issues ... 163
23.10.2.1. Printing ... 163
23.10.2.2. Font Fall-Back ... 163
23.10.2.3. Closing ... 164
23.10.2.4. Control and Alt ... 164
23.10.2.5. Windows Version 11 Problems ... 164
23.10.2.6. Minor Issues ... 164
24. Windows Considerations ... 165
24.1. Windows Version Support ... 165
24.2. Refreshing Windows ... 165
24.3. Do Not: Use Unicode UTF-8 for worldwide language support ... 165
24.4. High Contrast Theme ... 165
24.5. Multi-Monitor Environment ... 165
24.5.1. Mouse Click on Lime Window Ignored ... 165
24.5.2. Lime Does Not Appear at Start up ... 165
24.6. Gestures on Windows ... 166
24.7. Windows Font Smoothing ... 166
24.8. PDF Printing ... 167
24.9. Ctlr+AltGr ... 167
24.10. Bad Midi Devices Crashing PC ... 167
24.11. File Name Problems on Windows ... 168
24.11.1. Exotic File Names (fixed in 9.16) ... 168
24.11.2. Leading Spaces in File Names ... 168
24.12. Dialog Appearance ... 168
24.13. Right to Left and East Asian Languages ... 169
24.14. Lime for Other Languages ... 169
25. Lime Lighter ... 170
25.1. Manual Scrolling ... 170
25.1.1. Arrow Keys ... 170
25.1.2. TAB and Return Keys ... 171
25.1.3. Hearing ... 171
25.2. Automatic Scrolling ... 171
25.3. Full Screen Mode ... 171
25.4. Ticker-Tape and Autocue Contexts ... 172
25.4.1. Mark-Ups ... 172
25.4.2. Switching Between Normal and Ticker-tape or Autocue ... 172
25.5. Mark-Up ... 173
25.5.1. Mark-Up Zooming ... 173
25.5.2. Mark-up Graffiti ... 173
25.5.3. Printing Marked Up Pieces ... 173
25.6. Lime Lighter Specific Short Cuts ... 174
25.7. Lime Lighter Clicking & Gestures in Music Window ... 174
25.7.1. Configuration of 1 & 2 finger Tap Gestures ... 175
25.7.2. Press and Tap Gestures ... 175
25.8. Lime Lighter Preferences ... 176
25.8.1. Pedal Mapping ... 178
25.8.1.1. Old Infinity Pedal ... 179

25.8.1.2. Alternative Pedals ... 180
25.8.2. Lime Lighter Parameter Defaults ... 180
25.8.3. Lime Lighter Arrow Keys ... 180
25.9. Lime Lighter on Macs ... 181
25.9.1. VMware Virtual Machine ... 181
26. Lime Aloud ... 182
26.1. Lime Aloud Preferences ... 182
26.2. Toggling Lime Speaking ... 183
26.2.1. Lime Laconic Lyrics ... 183
26.3. Hearing Status ... 183
26.3.1. Hear Annotations ... 184
26.3.1.1. Lyric Spelling ... 184
26.4. Progress Messages ... 184
26.5. Navigation Short-Cuts ... 184
26.6. JAWS Issues ... 185
26.6.1. Dialogs ... 185
26.7. Alternatives to JAWS ... 185
27. Lime-GoodFeel Brailling ... 186
27.1. Lime-GoodFeel Preferences ... 186
27.2. Braille Window ... 186
27.2.1. Braille Font ... 187
27.3. Brailling Timeout ... 187
Appendix A Midi Percussion Key Mapping ... 188
Appendix B Music Font Character Mapping ... 189

1. Introduction

In the interests of releasing improved versions of **Lime** (**Li**ppold's **M**usic **E**ngine), 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 documentation. It describes enhancements and changes up to and including version **9.19.3**. Lime 9.19 is just a continuation of 9.17.

In general, with the exception of annotation anchoring, the use of Lime 9.15 and later is upwards compatible with earlier versions. The last 9.16 release was 9.16.99. Unless otherwise indicated, this recognizes most features referred to in this document as 9.17 or 9.18.

1.1. Summary

Main differences from 9.0x (amongst over 1234 enhancements and fixes) are:

- New file format to support the new annotation anchoring capabilities and larger zoom sizes. The 9.15/9.16 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**. Lime 9.17 introduced further enhancements to the file format; the 9.17/9.18 format can be read by 9.16. Lime 9.19.1 introduced further enhancements to the file format; this format can be read by 9.17 and later.
- 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. When dragging an anno, if Alt is pressed, it'll be high resolution.
- Batch update facility, enabling multiple files to be updated or to be changed to templates.
- Major improvements in MusicXML import. Improving MusicXML importing is a never-ending project. However export is still fairly basic.
- Fixer contexts to facilitate correcting imported and scanned scores and resolving other issues.
- 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.
- Panorama (ticker-tape) view contexts.
- Ability to transpose whole parts, including key signatures (useful for transposing instruments).
- Improvements to Lime Lighter, including full screen display, ticker-tape & autocue contexts, automatic scrolling, go to bar touch screen navigation, and a Low Vision template for display.
- Integrated support for polymetric and polytemporal scores.
- **9.17 and later has full Unicode support and always uses it**. 9.15.8 and later had the ability to edit Unicode annotations saved by post 9.16 Lime. 9.16 created Unicode annotations if needed.

1.2. System Requirements

1.2.1. Windows

Lime 9.17 - 9.19 is available for Windows XP and later; it is a 32-bit program (though a 64 bit version has been tested – contact us if you need it). It has been tested up to Windows 11. *Whilst it still works, support for Windows 2000 is deprecated*.

1.2.2. Apple Macintosh

Lime 9.17 - 9.19 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. Catalina Warning for MAC Users

Apple has stopped 32 bit applications, such as Lime, working on MacOS 10.15 (Catalina). In order to continue using Lime, you can upgrade to 10.13 (High Sierra) or 10.14 (Mojave), but 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.

High Sierra is needed if you want to use external MIDI devices, such as a keyboard, because Apple's Mojave does not handle them properly.

N.b. It is, however, still possible to run Lime on MacOS 10.15 or later by using a virtual machine (e.g. ***VMware Fusion*** or ***Parallels***), running Windows or an older MacOS. Google "*32 bit Apps Mac*" for more information. It is worth noting that WINE wrappers, such as **Porting Kit** (free) or **Crossover** can be used to run the basic Windows' version of Lime on any Macintosh that runs on Catalina or later. Email Lime.uk@virginmedia.com for WINE advice.

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, which you are strongly recommended to do.

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's basic facilities are available, whether or not you have donated.

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.

There are a lot of enhancements to annotations' support since Lime 9.15.

3.1. Annotation Selection – Right-Click

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

To drag an annotation, you must first select it, then right-click it again and drag it.

To select an annotation, the mouse must be fairly close to one of the end points of the annotation. If the *Control* (on Macs, *Command*) key is pressed on the keyboard , when the mouse is clicked, it must be very close to one of the end points. This can be useful to distinguish annotations very close together.

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

As of Lime 9.17, Unicode annotations are fully supported (entailing a minor change to the file format). When reading an older file, Lime 9.17 and later will convert all annotations to Unicode if possible.

9.15.8 and all later versions are capable of handling files containing Unicode annotations (they can recognize, display and edit them). In 9.16, when editing an annotation, if the text could not be rendered without Unicode, it was converted to Unicode. However, by default, if it could be rendered without Unicode it was. This also applied to annotations in MusicXML or NIFF imports. As part of the transition to full Unicode support, in 9.16.5 and later, there were facilities convert the piece to use Unicode (whenever possible) or otherwise.

In Lime 9.17 and later, Unicode is used whenever possible, because Unicode is the only reliable, compatible encoding. Essentially this means Unicode is always used except for annotations using music or symbol fonts. Unicode is also used for part and context names.

3.2.1. Code Pages (and conversion to Unicode)

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. Furthermore, on Windows' Lime since 9.15.8, for compatibility, before 9.17, if a non-English locale for entering text (e.g. Greek, Russian, Hebrew, Arabic, etc) was used, if it could be rendered using a valid code page, it was. If it could not be rendered using a code page, or if multiple languages were used in the same anno, Unicode was used.

Unfortunately there was an '*undocumented feature*' in Windows' Lime prior to 9.17. If a text annotation (e.g. title, composer, etc) were entered using a simple text assistant dialog (e.g. title, composer, etc), the character set was set to the default. The annotation would use the default locale code page of the host machine. If the annotation were basic ASCII, there was no problem. However, if it were foreign text (Greek, Russian, Hebrew, Arabic, etc), it would display fine on any Windows computer that defaulted to that character set, but not on any other (and not on Macs). The same problem applies to part and context names.

When converting such annotations to Unicode, Lime 9.17 and later needs to be told what the originating character set was. Otherwise they can become garbage. This applies to both Windows and Macs.

Provided warnings are turned on (or if no default has been specified in the preferences), Lime's warning dialog for older files includes a drop-down choice of the code page to use for any annotation for which the character set is not specified[1]. The default is the code page for the host machine.

If a file is opened within Lime using *File->Open…*, if the Shift key is pressed when selecting the file, older file warnings will be forced on for the selected file, thus allowing its locale code page to be specified.

If the result of opening an old file has garbage titles, etc, then one should force warnings on (see general preferences, section 16.1.2) and experiment. Note that other annotations, such as lyrics, entered directly, should carry any appropriate charset and are probably not, therefore, affected.

The ***Batch…*** update facility (*see section 22.49*) also includes the option to specify the default code page for the files being updated. If you have pieces that originated on Windows and relied on the computer's code page, it is recommended that these pieces be updated (if possible, on Windows[2]) to 9.17 and Unicode. As well as being compatible with any Windows system, they will then be fully compatible with Macs.

3.2.2. Pre-Unicode Fonts on Old Macs

Prior to Unicode, older Macs used special Fonts for foreign character sets, such as Japanese. The fonts used were remembered in the Lime files, so the font could be used for the character set. These character sets were often the same as the equivalent ones on Windows. When Lime 9.17 or later reads such an old file on a Mac, the original font and hence character set should be identifiable; in the unlikely event that it isn't, it may be necessary to specify the default character set to use when opening it[1].

On Windows, however, if reading such an old file that was created on a Mac, with the exception of the *Osaka* and *Hiragino Mincho* Japanese fonts, the special fonts are unrecognizable, so Windows' Lime has to be told what the appropriate code page is. This is possible with 9.17 and later, so most such old Mac files can now be read on Windows.

---

[1] There is an option not to convert to Unicode, but it is <u>not recommended</u> and should only be used if all else fails. It is at the bottom of the list of default code pages; it **must** be selected with the **shift-key**, otherwise it is ignored. *Feel free to contact us, if you have character set issues updating old files to 9.17 or later.*

[2] Lime 9.17 on Macs can handle annotations using old Windows character sets, however there are known to be some conversion issues.

3.2.3. 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). ***Arial Unicode MS***, which is available on many modern systems, is a useful font for multi-lingual text.

Because of its antiquity, it is recommended that the *Symbol* font is not used, unless absolutely necessary. There should always be appropriate unicode symbols in a normal font.

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 and 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

Anchor	***Relativity***	***Description***
Note	HEAD	The anchor point is the left edge of the note-head. This is the default.
	Left	The anchor point is just to the left of any accidentals, parentheses, etc 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.
	Right	The anchor point is just to the right of any dots, parentheses, etc on the staff chord in which the annotation's note participates. *Available from 9.17.5, but recognized from 9.16.8*.
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.

Anchor	*Relativity*	*Description*
	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

Anchor	*Relativity*	*Description*
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	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 that are relevant to all parts in a system. It is available in 9.16.9 and later.

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; it applies to both text and line/curve annotations.

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[3].

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:

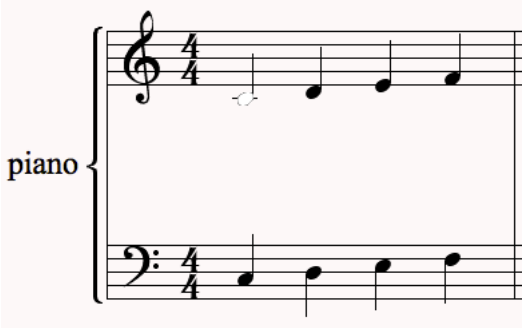
- The default separation specified in the context layout…
- Adjustments made by staff/system separation parameters.
- Vertical compression of the page if it is too dense to fit.

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.

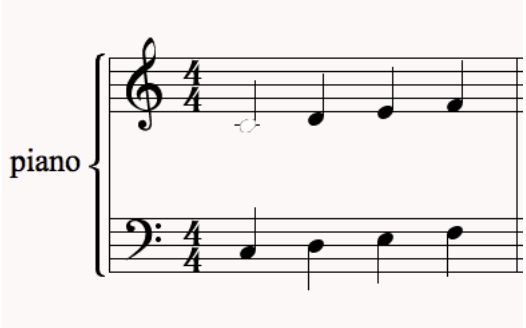
A common use of staff spacing adjustment is to place labels on grand staves. If such a label is placed on the upper staff, positioned mid way between the two staves, and the *Staff Spacing Adjust Style* is set, it will always remain roughly at that midway position, whatever the actual separation.

---

[3] The scaling uses the context layout values, any relevant parameters, and any compression necessary to fit onto the page. It does not include any Staff Drag. So Staff Drag may adversely affect the result.

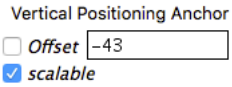


Unscaled, 80pt separation



Scaled, 60pt separation

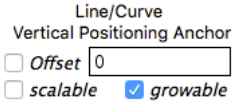
In the *Annotation Placement* dialog, the vertical offset is always the unscaled value.



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 its 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

As of 9.16.9, 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:



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, for example, 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 (draw line between the middle of one staff to the middle of the next, then adjust the vertical clipping at either end so that the visible portion is only between the staves).

3.3.2.5. Extended Line/Curve

If a line or curve is extended, cannot be vertically anchored to the page. If an attempt is made to vertically anchor it to the page, it will be changed to being staff anchored.

3.3.3. Text Justification

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

Because justification and anchoring are closely related, text justification is specified in a unified dialog[4] along with the anchoring, which is invoked using the '*Annotation Placement…*' menu item from the *Annotation* menu (see section 3.3.5).

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

3.3.3.1. *Horizontal Locking*

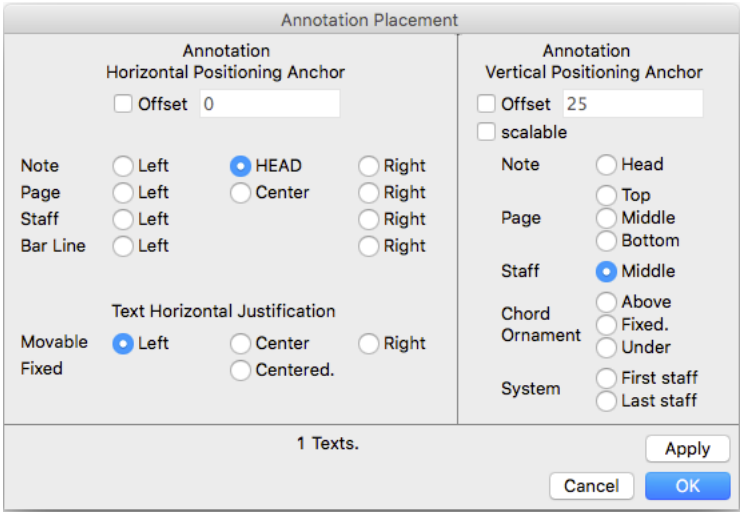
There is also a special justification, known as '***Centered***', which not only centres the annotation at its horizontal anchor[5] (see section 3.3.1), but also prevents it being moved horizontally. Its main use is for lyrics.

3.3.4. Leave Horizontal Space

The *Annotation->Style->Leave horizontal Space*, option specifies that extra space will be left in the music to avoid overlap with the annotation. This is normally used with lyrics. It only applies to annotations that are horizontally anchored to their note's note-head. It has no effect and cannot normally be applied if the annotation is not note-head anchored.

3.3.5. 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 on Windows **Control-,** (comma) and on Macintosh **Command-,**.



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.

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

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.

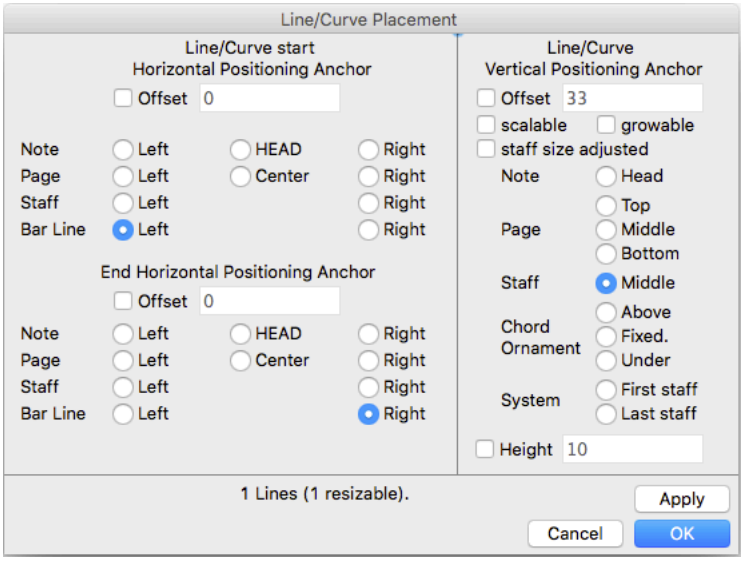
If an offset value is changed, it will be set for all selected annotations. This can be useful as an alternative way of aligning annotations.

3.3.6. Line/Curve Placement Dialog

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

The dialog includes the ability to set the height of the Line(s)/Curve(s). This is the vertical distance from the left end to the right end; a value of 0 means a horizontal line, which is an easy way to ensure a line is truly horizontal. The height value can only be changed if the associated check box is checked. Similarly, if the line/curve is not extendable, its length may be specified; a value of 0 means a vertical line; which is an easy way to ensure a line is truly vertical.

The dialog also a ***staff size adjusted*** check-box option. If this is checked, the vertical location will be adjusted if the staff size changes from when it was set. It is equivalent to the ***Staff Size Adjust*** text style, but only affects the vertical position of the beginning of the line/curve; it does not affect the height.



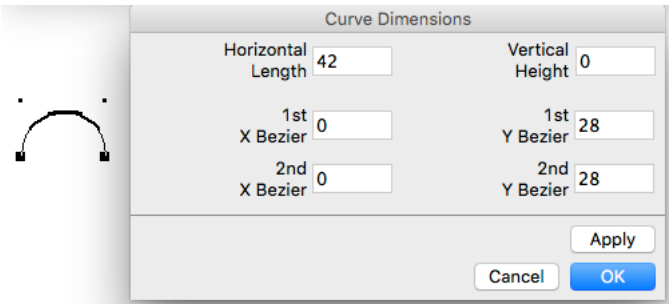
Line/Curve Placement Dialog

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 then one line/curve is selected, changes to the End Horizontal Positioning only affect the resizable ones (if any).

3.3.6.1. *Curve Dimensions*

In Lime 9.19.1 an esoteric dialog now allows one to specify the actual Bezier control point positions of a normal curve. In the Line/Curve Placement dialog, if just a single curve is selected, if you Shift-click the ***Height*** checkbox, a sub-dialog will be brought up allowing one to specify the position of the 2 Bezier control points. The 1st point is relative to the start of the curve and the 2nd point is relative to the end of the curve (X and Y are the horizontal and vertical values).



Curve Dimensions Dialog

The example specifies an upward semi-circle, as shown (in this case, curve is fattened).

3.4. Text Editing

Whenever a text annotation is being edited, the selected note will be flagged but not actually flashing,

3.4.1. Editing Existing Annotations

When a single text annotation is selected, it can be edited by **left**-clicking on it (without *Alt*!) or pressing *Return*. If the *Shift* key is down, all of the text will be selected and anything typed will replace the whole lot.

There are some minor issues with sizing when editing complex annotations (particularly chord symbols) in older files using backslash expressions. Because of this, unless the *Shift* key was used, if no changes were made at all when editing, the annotation will be left completely alone.

3.4.2. 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 left 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 ***Tahoma***[6] 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 *Tahoma*, but in a few cases *Arial Narrow* 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, typically *Tahoma*.

Even with modern Lime files, it should be noted that if the file was created on a Mac, using in-built exotic fonts, with the exception of Osaka and Hiragino Mincho, Windows will probably not recognize them (and, if pre-9.17, the relevant code page will need to be specified – see section 3.2.1).

3.6.1. Replace one Font by Another

Since 9.17.9, there is a facility to globally replace all use of a font by another. This applies to all annotations and explicitly specified parameters in the whole of a piece. This facility is in *File-->Options for Piece…* It is invoked by clicking the *Replace one Font by Another button*.

If you click the button, you will be prompted for the font to replace and then the font to replace it by. The special Lime music or symbol fonts cannot be replaced to used as a replacement.

Apart from changing the appearance, the facility can be useful to replace old, *Type 1* fonts.

---

[6] Prior to 9.17.9, *Times New Roman* was used as the substitute. Since 9.17.9 ***Tahoma*** is used, because there is more consistency in fall-back when an exotic glyph is used.

3.6.2. Font Names

On Windows, old Tufa and Marl fonts may have music looking names in the in-built font dialogs. E.g, Tufa may appear as and, less likely, Marl as

You should ensure the latest versions of the Marl and Tufa fonts are installed.

3.7. Backslash Codes in Annotations

Backslash expression 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]:

\	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 (~). In simple fingering, it keeps the space with the next character.
\\	Backslash followed by another backslash is a single backslash. This allows a backslash to appear in an annotation.
\N	Backslash followed by N (capital-N) is a new-line. Text following it will be on a new line (see section 3.8).
\R	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.
\;	Reverts font size and style to that in force at the beginning.
\>>	Decrease font size by 4 points; replaces \1. (\1 is now number 1 in a music font).
\<<	Increase font size by 4 points; replaces \2. (\2 is now number 2 in a music font).
\>	Decrease font size by 1 point.
\<	Increase font size by 1 point.
\<>	Sets the base font size (for subsequent \>, \<, etc) to that of the last character.
\,	A thin-width (non-breaking) space.
\.	A dot (as in a dotted note).
\..	A double dot.
\+-	A quarter sharp symbol
\+-+	A three quarter sharp symbol
\-+	A quarter flat symbol
\-+-	A three quarter flat symbol

\\\|	A vertical line
\!	A large slash (e.g. as used for slash rhythm note heads and beat repeat symbols)
\`	A slash (e.g. as used in figured bass)
\l	Toggle underline (like \b and \i toggle **bold** and *italic*).
\v	Toggle $_{\text{subscript}}$ (like \u toggles ^{superscript}).
\FONT	Reverts to the base font (e.g. after \MARL).
\niente	A niente circle (e.g. for diminuendo to nothing).
\arp	The Arpeggio character. Two arpeggios with a line feed between them (`\arp\N\arp`) will be a big (tall) arpeggio character.
\arpup	An Up Arpeggio – one with an upwards pointing arrow (in 9.17 and later).
\arpdown	A Down Arpeggio – one with an downwards pointing arrow (in 9.17 ad later).
\0 .. \9	\ followed by a digit means the equivalent digit in a music font, as used, for example, for numbering full bar rests.

While editing, the base font size effective at the start of the annotation is shown in the Piano window. The backslash sizing codes (\<<, \>>, \<, \> and \<>) start with reference to this value. For example, if the base size is 12pts, the back slash expression '**\<<;\f;**' will result in a 16pt *forte* symbol. Note that the sizing codes 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 spaces).

In addition, Style->*Fingering* can be used to set or clear vertical stacking and, optionally, to set or clear fingering classification[7] 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

Lime 9.17 and later has 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 5[8];
- 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 **c d e f g a b**)

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 Lime Aloud and/or brailling purposes. If anything more sophisticated is needed, ordinary annotations, classified as fingering, can always be used.

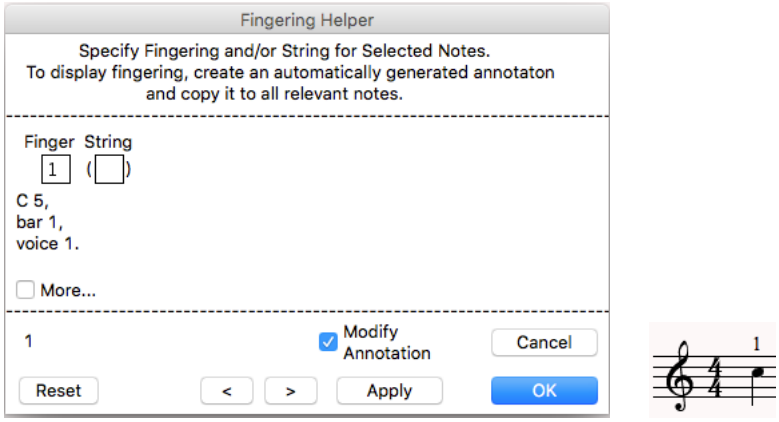
---

[7] Note that programs such as GoodFeel require the classification to be ‘***Fingering***’ in order to handle fingering annotations properly.

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

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.



Basis use of the Fingering Helper

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

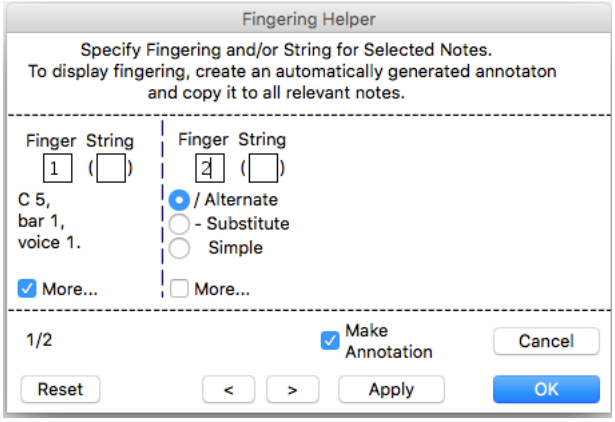
If the note is actually part of a staff chord, whose other notes also specify fingering, then the resultant annotation can show the fingering for all notes in the conventional stacked manner. When making auto-generated annotations, one has the option to specify whether the annotation is generated from just the note, or from all or part of the chord (see section 3.10).

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.

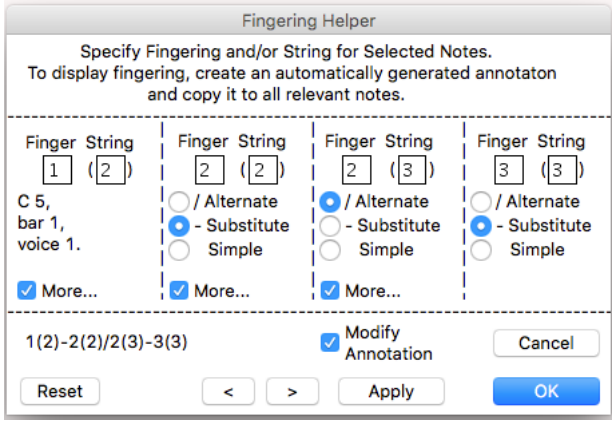
The fingering helper dialog only allows valid fingers and strings to be used, as defined above. Any attempt to enter more than one character or an improper one, will be rejected (with a beep).

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 of voices on the selected staff. 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 n an annotation is shown above the *Reset* button.



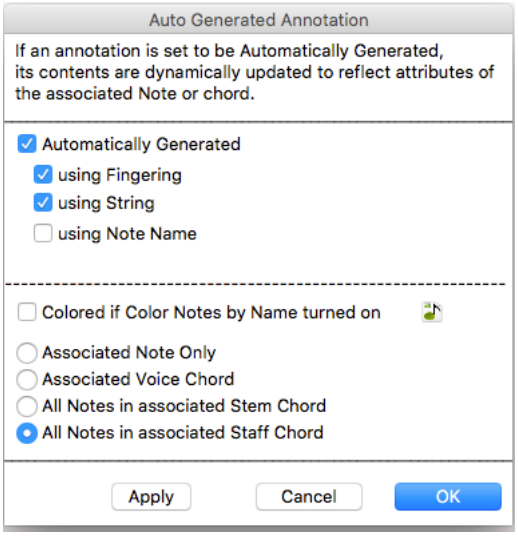
Alternate Fingering specification



Extremely Complex Fingerings and Strings

3.10. Auto-generated Text Annotations

Lime 9.17 and later has the ability to define text annotations to be automatically generated from data about the note with which they are associated. An annotation can be specified as Auto-Generated using the *Annotation->Auto Generated…* menu option.



Creating an Auto-Generated Annotation

If specified as *Automatically Generated* (check box). The possible data (check boxes) are:

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

Note name. The *Note Name* is the basic name of the associated note and is one of **CDEFGAB**; it can be useful for education purposes. In or after Lime 9.18.1, if the checkbox is fully checked, the name will be followed by the accidental (if not natural) and, optionally (another checkbox), the octave.

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.

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 9.16 and earlier will treat them as ordinary fingering annotations.

3.11. Multi-Line Text

Support for multi-line text (created by using **\N** 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 (***\N***) 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 annotations[9]. 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.

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

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.

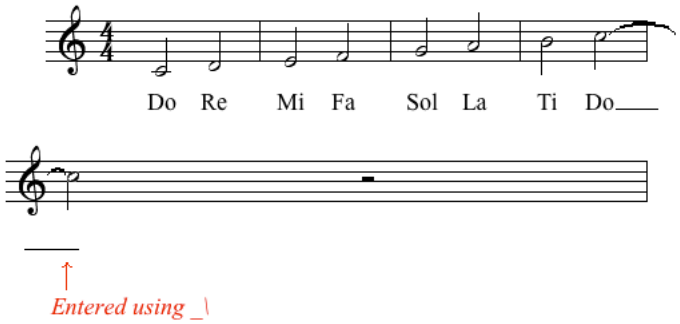
3.13.1. Terminating Underline Extension

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).

Currently (9.16), if the extension goes over a system break, it 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 space[10] 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!

[10] 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. Annotation Assistants

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

When a text annotation assistant is used, anchoring will be appropriate to the type of annotation. Moreover default positioning will be applied. The user may need to adjust annotations' positions if the result overlaps with another annotation.

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 chord and fret grid 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.

3.15.1.1. 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♭9) *Old sizing*: Cmaj(♯5♭9)

3.15.1.2. Chord Symbol Naturals

Normally in a chord symbol, the chord name's accidental is only included if it is sharp or flat. However, there is a checkbox in the chord symbol dialog to '***Show Naturals***'. If this is checked, a natural symbol will be shown, if it is not a sharp or flat chord. This is unusual, but is sometimes used.

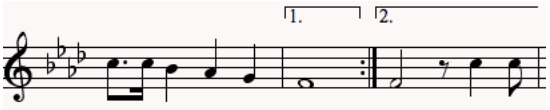
Normal Chord Symbols: Gm F Chord Symbols showing Naturals: G♮m F♮

Normally a chord symbol should be set as interpreted, so that it will be transposable if necessary. Unless it is not interpreted, if any change is made to it, Lime will attempt to normalize it to keep it transposable and may remove superfluous characters immediately after the main or bass chord letter. If you want to add other accidentals or symbols immediately after the main or bass letter, the annotation style should be set to '***Do Not Interpret***'.

3.15.2. Line Bracket Assistant

As of 9.17.9, the menu option *Annotation->Line and Curve Build Assistants->Ending Bracket…* allows one to create a bracket and associated text to indicate a 1st or 2nd ending round a bar line that has left dots. To create a 1st ending bracket, select the first note in the first ending bar, then invoke the menu option. To create a 2nd ending bracket, select the first note after the repeat bar line, then invoke the menu option. The option dialog allows one to specify which sort of bracket is needed and, moreover, whether to set the associated bar line "special for hear option". Note that a 1st ending can only be created if the selected bar is no more than a few measures before the left dotted repeat bar.

Example of 1st and 2nd Ending Brackets:



3.15.3. Other Line/Curve Assistants

See section 22.51.2 (includes drawing a Phrase Indication curve, see section 3.17.2.2).

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. You should assign an appropriate category to all Text annotations.

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* categories.

3.16.1. Text with Marl+defaults

This 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.

This alternative menu is primarily for creating new annotations, rather than changing an existing one. Changing an existing annotation in Times font to Marl font is usually not a good idea.

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. Note that the \N new-line facility enables a single multi-line figured bass annotation to be created.

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. Line Types

As well as straight lines, hairpins and brackets, Lime now offers rectangles or parts, thereof and arrowed lines.

If you have a Line selected and click on one of the Line types in the *Annotation->Line* menu, you will be asked whether a new Line should be created or the selected Line changed.

3.17.1.1. Rectangles

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.

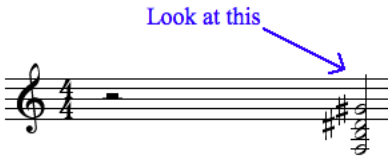
Like other 'meta-lines' (hairpins, etc), rectangles' selection points are the top-left and bottom-right. However, you can also select them by right- (or alt-) clicking on the bottom-left or top-right.

3.17.1.2. Arrowed Lines

Lime 9.17.6 and later supports arrowed lines[11]. These are lines with an arrow at one end or both ends. Arrowed lines are specified using one of the *Annotation–>Line* options. An arrow is right-angled and the length of a line arrow is a parameter. The actual size of a line arrow is the parameter value plus the line width.

Using the *Annotation–>Line* options, one can specify an arrow at the start or end of a line. The start of a line is the end used for anchoring it. Unless the line is vertical, this is always the one on the left side. If the line is absolutely vertical (zero length), the anchoring will depend on how it was drawn. If, after drawing or changing a line with a single arrow, the arrow is at the wrong end, then you will just have to swap it.

Note that the *Annotation Placement…* dialog, shows the (horizontal) length and (vertical) height of a line. If the length is zero, then the height indicates its direction; if negative, the start is at the top, else it is at the bottom.



Example of use of an Arrowed Line

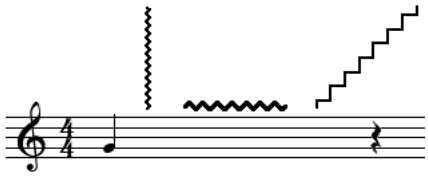
3.17.1.3. Dashed Lines or Curves

A Line or curve may be dashed. This is done by specifying the dash and gap lengths using the menu option *Line and Curve Style >Dashed or zigzag…* Note that Lime will tweak the specified dash and gap lengths to ensure that a dashed line always starts and end with a dash.

[11] Arrowed lines are recognized by Lime 9.17.5 but not by any earlier version.

3.17.1.4. Zigzag Lines

Lime 9.17.6 and later supports zigzag lines[12]. These are lines drawn as a right-angled saw-tooth. Some of examples are shown in the following figure:



Examples of Zigzag Lines

A zigzag's dimensions are specified using the *Line and Curve Style >Dashed or zigzag…* menu option. In the resultant dialog, check the Zigzag option and specify the length. The length is the approximate distance between two teeth on the same side of the line, excluding the line width. The actual distance may be slightly shorter as Lime ensures that there are an integral number of teeth. The size of a tooth is proportional to its length (length/√2).

Only lines can be zigzag, it is not possible to have a zigzag curve. Note, also, that a zigzag line cannot be dashed as well. In the unlikely event that a zigzag line carries an arrow, the arrow size will be increased a bit.

3.17.2. Curve Types

The basic form of a Lime curve is like a slur (which is its primary purpose). A curve is essentially a Bezier curve and the control points can be used to adjust the slur shape or create all types shape.

3.17.2.1. Ellipses and Circles

It is possible to create an ellipse or circle using two Bezier curves (it cannot be done with one curve). However this is tricky (see section 3.3.6.1), so as of 9.19.1, ellipses[13] and circles are available as a type of curve. A circle is essentially an ellipse, but its proportions are fixed. If you resize an ellipse, its proportions can change, however if you resize a circle, it will remain a circle.

To create an ellipse or circle, select it in the *Annotation->Curve* menu. You can convert an ellipse into a circle or vice versa. This is done by selecting it and then clicking on the type in the menu. You cannot convert an ordinary curve to or from an ellipse/circle. In the *Annotation->Curve* menu, the *Curvy* item simply indicates that a selected curve is an ordinary curve (not an ellipse or circle), or an ordinary curve is selected for creation. The Curvy item is never actually activated – use the *New Curve* item to select an ordinary curve for creation.

3.17.2.2. Phrase Indicator

A simple curve (not an ellipse or circle) may be tagged as a ***Phrase Indicator***. Currently this has no effect whatsoever on its appearance or behaviour. Its purpose is to be used by Lime-GoodFeel to identify the indicated music phrase, when brailling. See section 22.51.2 on the assistant for creating one.

---

[12] Zigzag lines are recognized by Lime 9.17.5, but not by any earlier version.

[13] Ellipses and circles are recognized by Lime 9.19, but not by any earlier version.

3.17.3. 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.

3.17.4. 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 of a line to change the appearance. If the annotation is a curve and the shift key is used, *Curve Inversion* will be done (see below). If the annotation is an ellipse (or a circle) nothing is done.

3.17.5. Curve Inversion

Curves can be inverted using the *Annotation-> Curve* menu. Inversion simply flips the curve's Bezier control points. Typically inversion will convert a curve-up into a curve-down, and vice versa (this is the primary purpose of the facility); it will also convert a left facing curve into a left facing one. If the curve is an ellipse (or circle) the appearance does not change, however the right selection point will be inverted. 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.

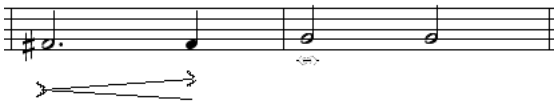
3.17.7. Line Staff Size Adjusted

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.

3.17.8. Resizable (Extended) Lines and Curves

A Line of any kind or a Curve can be extended by first selecting it, then clicking on a Note (in the same voice, but can be on the next page) to which one wants it extended. The Line/Curve will remain partially selected, indicating its extendibility. After that, to do the extension, invoke the '*Edit->* ***Extend Line or Curve to Here***' menu option.



Example of a Partially Selected (wee right arrows), Extendible Hairpin

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 or phrase curves, 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[14]. The position can, of course, be changed by dragging the end of the line/curve; and the anchor can be changed using the '*Annotation->**Annotation Positioning***' dialog. The end is initially horizontally anchored to its note.

If a line/curve style is set to Extended using the *Annotation->Line and Curve Style* menu, the end of the line/curve will not move. The end will become horizontally anchored to the note; but the beginning will remain anchored as before.

Like text annotations, the anchoring for a line or curve 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 itself[15], the duplication can become undesirable; in which case, it may be necessary to use two separate curves.

While there is a visible extendable line, it is partially highlighted with a wee right arrow at either end (see figure above). It is extendable until another annotation is selected, or any other change to the piece is made. Its highlighting can be removed, by double clicking its mode button in the Piano window; then it cannot be extended using '*Extend Line or Curve to Here*' until it is selected again.

3.17.8.1. 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 note.

3.17.8.2. Clearing Extendibility

A resizable/extendable line/curve may be converted back to being un-extendable, 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 back to 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.

---

[14] If the extend to here note's position is to the left of the line's original start, the result can be strange due to flipping.

[15] A 'complex curve' is one whose 1st Bezier control point is to the right of its 2nd one.

3.17.8.3. 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.

3.17.8.4. 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.

3.18. 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. Duplicate Annotations *at Every System / on Every Page / in Every Part Context*

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.

Annotations that have been duplicated *at Every System* or *on Every Page* or *in Every Part Context*, are flagged as such, and can be cleared using the appropriate *Edit->Clear->* menu item.

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 manual[16]). Common uses of *Duplicate in Every Part Context* include duplicating tempo markings, rehearsal signs and titles so they appear in all parts' contexts. Note that duplicating (or clearing) into every part, can be performed only from the Score context; in other context the option is not available.

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

Notes If one or more notes are selected, then **all** 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 ones remain.

3.19.1.1. Invisible Annotations that have been Duplicated

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 Annotations that have been Duplicated to Every Something

If an annotation, that has been duplicated *at Every System* or *on Every Page* or *in Every Part Context*, is deleted (cleared), 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), all duplicates are deleted with no warning.

Annotations *Duplicate in Every Part*, 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.

---

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

3.19.3. Modifying Text Annotations that have been Duplicated

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 deletion[17], in the case of *Duplicate in Every Part* annotations, **propagation only happens if the change is made in the score context**.

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. 1st Anno, on or after Note

Because annotations on a note are kept before the note. The option to select the next annotation will actually select the next annotation after it. It has always been like that. The new menu option **Annotation->1st Anno, on or after Note** will start looking immediately after the previous note (or start of piece) and thus, if there are any, the first annotation on the selected note.

3.21. 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 selected[18]. 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.

3.22. Shoving Annotations

As of 9.18, if one or more annotations are selected, they may be shoved about on the page using ***ctrl/cmd+alt+*** an ***arrow key***[19]. They will be moved by a little bit in the direction of the arrow key; if the *shift* key is also used, the resolution will be very high.

---

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

[18] This was the behavior prior to 9.16.9, even when the shift key was not used.

[19] On Windows, the JAWS screen reader might prevent *Ctrl/+Alt+ arrow key*, if so JAWS is running, shoving will not be available. If this is a problem, contact Dancing Dots to get an up to date version of Lime JAWS' scripts.

Successive use of any of the arrow keys, with ***ctrl/cmd+alt***, is treated as a single change when it comes to undoing, providing there is a reasonably short time between each use. It is worth noting that if the keys are held down, there will be successive shoves (nudges).

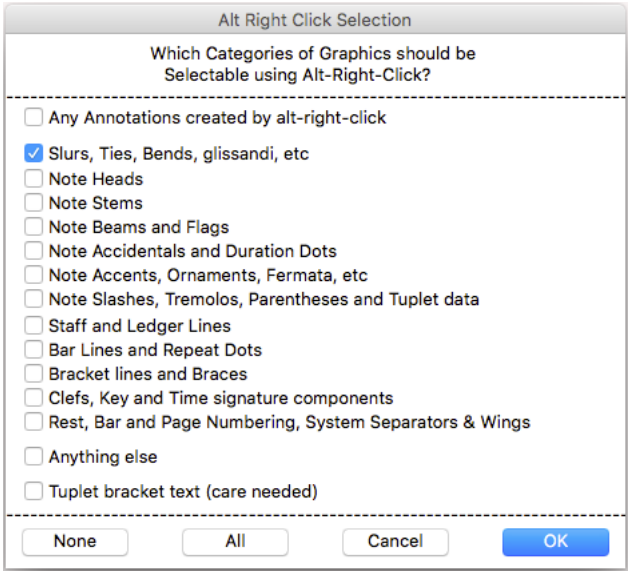
For compatibility, if no annotations are selected, ***ctrl/cmd+alt+*** an ***arrow key*** is treated like a simple arrow key and will go to the next/previous note/voice.

3.23. 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.

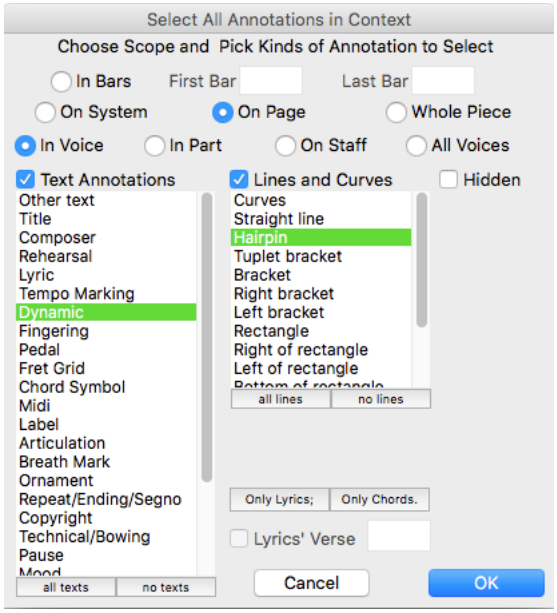
Group selection is also possible with ***Alt+right button***. If group selecting with *Alt-right-button*, normal annotations will be ignored (unless the shift key is down as well).

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*.

3.24. Annotation Selection by Class

There is a facility (after 9.18.2) for restricting the classifications of annotation that should be selected when using the **Annotation->Next** or **Previous Annotation** menu options (or Alt+ right/left arrow). By default any annotation is selected, but it is often useful to restrict to certain categories; for example one might just want to select lyric text annotations or just hairpins. The required classification can be chosen using a dialog shown by the **Anno Selection Classes** menu option.



Example of Choosing Annotation Selection Criteria

An example of the dialog is shown in the figure above. In this case selection will be restricted to dynamics and to hairpin lines. This unusual example means that both text and line classification lists are shown; they are only shown if the associated checkbox is checked.

The **Reset** button will reset the choices to the default, namely all texts and all lines and curves. Each list box has an '**all**' and '**no**' button to select all or no items in their list. If the check box for texts or lines and curves is checked with the shift-key, then all items will be selected.

The **Only Lyrics** button is a quick way of setting that only lyric text annotations will be selected; the **Only Chords** button is a quick way of setting that only chord symbol text annotations will be selected.

If lyric text annotations are selected, it is possible to restrict the selection to a particular verse. This is done by checking the **Lyrics' Verse** check box and entering the required verse number in the associated edit field. A value of zero implies chorus or lyrics, whose verse has not been set.

When group selecting next or previous annotation (using shift key), by default the criteria selected in this dialog only apply if an annotation is not already selected. If one or more annotations are already selected, the default is just to use the classes of those already selected – see section 3.21. If, however, the checkbox '***Group selecting, also apply if annos are already selected***' is checked, then the classes will be those specified in the dialog as well as any currently selected annotations.

If **Ok** is given, the choices made are remembered and are the defaults for next time the dialog is used.

3.24.1. Select All Annotations

The *Edit->Select All->Annotations* dialog is similar to Annotation Selection by Class, with the addition of radio buttons to choose scope and which voices (see section 22.24).

3.25. Remember Selected Annotations

If there are selected annotations, the *Annotation->Remember Selected Annos* menu option, will remember them, so that they can be reselected later or copied to another place, using one of the *Edit->Duplicate Remembered Annos->* functions (see section 3.26).

When multiple annotations are cut or copied to the clipboard, and then pasted, they will all be inserted on the same note[20] (and user is warned about this when cutting or copying them). The remembering facility overcomes this limitation. When *Remembered Annotations* are duplicated, they are inserted so they appear at the same places, relative to the *position of the chord to which they are duplicated,* as they did at their original place, relative to the earliest remembered annotation.

Annotations can be remembered between views and contexts on their piece[21], so one can remember a set of annotations, then go to another context or duplicate them there. If <u>any</u> of the remembered annotations' voices are not in the destination context, the voice to which they are all duplicated must be specified (in the duplicate annotations dialog). They can also be selected in another context, provided all their voices are in it.

There is a short-cut for the *Annotation->Remember Selected Annos* menu option:

- Windows *ctrl+<u>alt</u>-=*
- Macs: *cmd+<u>ctrl</u>-=* (on Macs, *cmd+alt-=* is usurped by the operating system, so *cmd+ctrl is used*).

[20] It has always been like that.

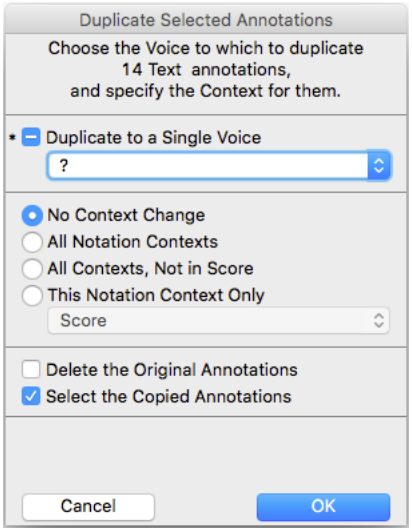
[21] Annotations are not remembered between pieces.

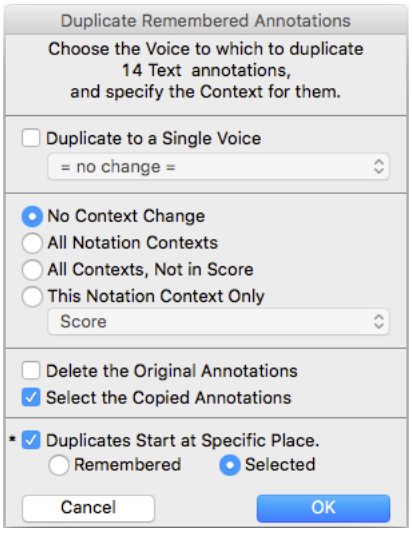
3.26. Copy (duplicate) Annotations *To a Specific Voice* or *a Specific Place*

If there are Selected or Remembered Annotations, the *Duplicate Selected Annotations->* or *Duplicate Remembered Annotations->* menus offer options to **copy** them to equivalent places (bar/beat) at a different place in the piece or just to another voice.

- ***To a Specific Voice***:
  Copy all selected annotations into a specific voice so that they appear at the same places as they did in their original voice. This option is available if there is more than 1 voice in the piece. For this option, you must specify the destination voice.
- ***To Remembered Place:*** *(this option is available if there is a remembered chord; see section 22.51)*:
- ***To Selected Place***: (*this option is available if there is a selected note)*:
  Copy all chosen annotations so that they appear at the same places, relative to the specified place, as they did at their original place, relative to the earliest selected annotation. When copying to a specific place, it is possible to specify that they are copied to a single voice, but the default is that there is no change and to copy them to their parent voice, providing the parent voice is active in the current context.

When one of the menu options is invoked, there is a dialog to choose how to duplicate.





Duplicate Annotations Dialogs

They are actually the same dialog, but the defaults depend on the duplication option chosen. The duplication that will happen depends on the ***Duplicates Start at Specific Place*** checkbox (the one near the bottom of the dialog); if not checked or not there, a specific voice must be specified. A partially checked ***Duplicate to a Single Voice*** checkbox means a voice must be specified, but hasn't yet been.

For all functions, there are the following options:

- Delete the Original Annotations after duplicating them.
  If this is chosen, you will be warned about the deletion. This is to avoid doing it inadvertently.
- Select the Copied Annotations.
  If this is chosen, the originals will be unselected and all the copied annotations will be selected. This is the default. It means one can quickly change their characteristics, without having to reselect them.
- Choose the contexts in which the copied annotations will appear. The default is that there will be no change. N.b. *This Context Only* means the current one; the drop-down identifies what it is.

3.26.1. Destination Place

The selected annotation, which is earliest in the piece, identifies the original reference place (its chord's duration from start of piece) for the duplication. The destination reference place depends on the option:

- *To a Specific Voice*:
  The equivalent place in the destination voice.
- *To Remembered Place*:
- *To Selected Place*:
  The offset (music duration) of the remembered chord from the earliest selected annotation's chord. The specified Chord must be exactly at the place, where one wants the earliest selected annotation to be copied.

If there is not an exactly equivalent place at the destination for an annotation (i.e. there is not a start of chord there), the copy will be placed on the chord that overlaps it. You can always move it if that is not what is wanted.

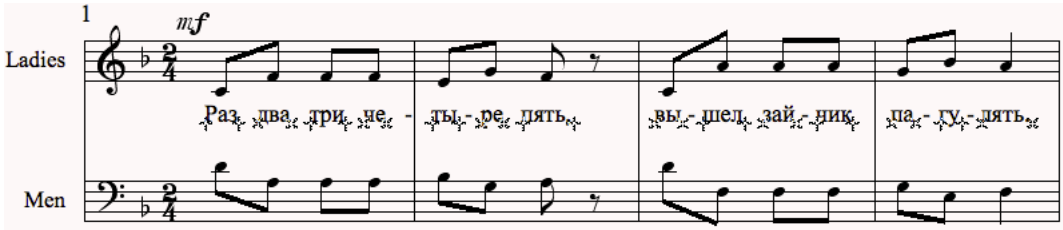
3.26.2. Selecting Annotations to Duplicate

The annotation that will be duplicated can be anywhere in the piece, e.g. originally selected using *Edit->Select All->Annotations in Context…* menu option (see section 22.24) or selected by hand. Wherever they are, they will be copied to their equivalent place.

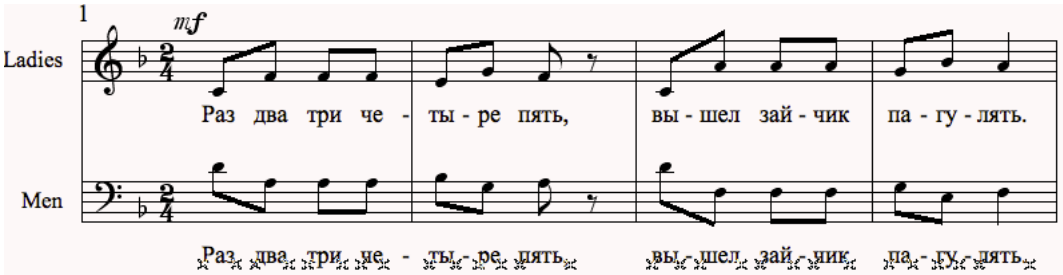
If an annotation's duplicate would be on the same voice and in exactly the same place as the original, then it is not duplicated.

3.26.3. Examples

3.26.3.1. To a Specific Voice



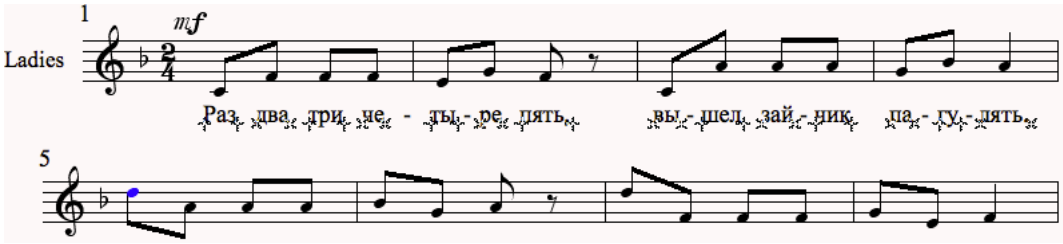
Original



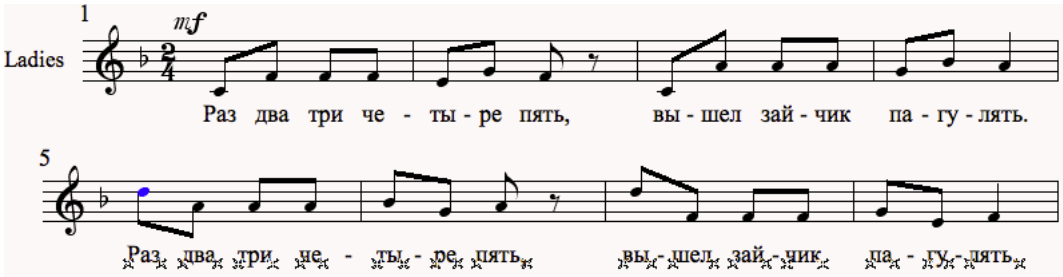
After Duplication to Men

In this example, the selected lyrics on the Ladies' voice were duplicated to the same place on the Men's voice.

3.26.3.2. To Specific Place



Original



After Duplication to Specific Place

In this example, the specific place was bar 5, beat 0 (blue note in the example). The lyrics were duplicated to start at that chord (on the same voice – Ladies).

3.27. Annotation Creation Mode=

The *Annotation Creation Mode=* options in the *Annotation* menu provide menu equivalents of clicking the mode buttons at the left of the piano window; see the **Annotation Mode** section (c. page 34) in the main users' manual.

They are checked (in the piano window, they are highlighted), according to which mode Lime is currently in. If more than one annotation type is selected, their types will be partially checked.

If just one annotation type is selected, its item will be partially checked if nothing is selected and, therefore, a new annotation could be being created.

The modes are ***Music*** mode, ***Text*** mode, ***Line*** mode and ***Curve*** mode. If *Music* mode[22] is selected, there are no annotations selected or being created.

[22] The *Music Mode* option is the same as the *Note->Note* menu option (see section 22.19.2).

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 and to play until the end of the piece.

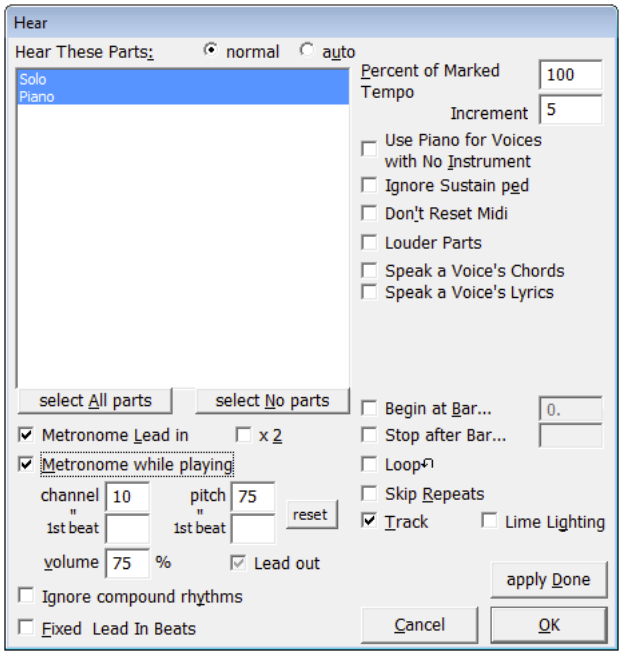
If play-back is active, *Control/Command-period* by itself will suspend play-back; if it is pressed again, play-back will resume at the beginning of the bar in which it was suspended. If used with the alt-key (option), play-back is stopped completely. If used with the shift-key the Hear dialog is always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

4.2.2.1. 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.

4.2.2.2. Metronome Accuracy

The metronome facility does assume that the piece's measure lengths do correspond to the time signature and there are not ones with spurious lengths, particularly just before a repeat, a go to or a loop (or at the place they go to).

4.2.2.3. Metronome Pitch and Channel

In both the *Hear* and the *Record* dialog, it is possible to specify the MIDI pitch and channel for the metronome. If channels 1-16 are selected, the metronome will be played on that MIDI channel on the output device; the recommended (and default) channel is the percussion channel, **10** – see appendix A for a list of percussion pitches - Lime's default pitch is 75 (claves). There is a ***reset*** button that gives one the option to reset the metronome's pitches, channels and volume to their default.

If channel 0 is selected, the metronome will be the current system 'beep' on the computer speakers.

4.2.3. 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 ad 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.

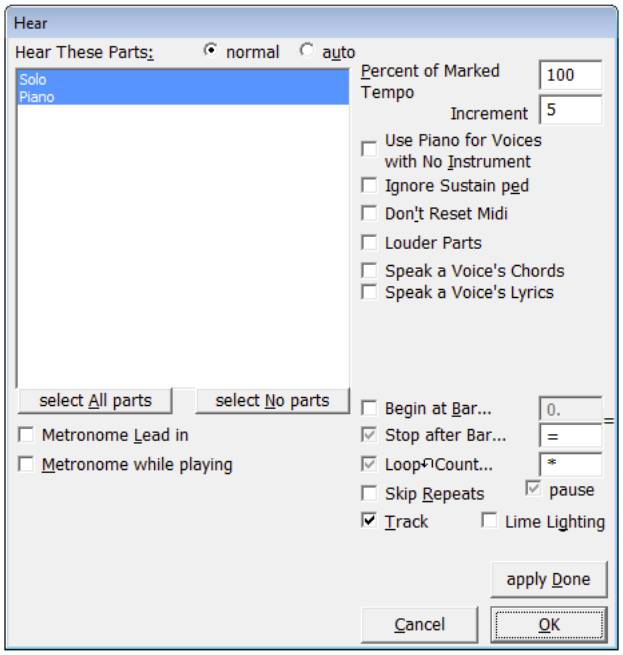
4.2.4. Skip Repeats

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.

4.2.5. Hear Selection and Looping

By default, when hearing is invoked from the Hear dialog, hearing will be just to hear from the currently selected chord until the end of the piece. As of Lime 9.19.2, there are options to specify where hearing should stop and where it should begin. There is also an option to play the selection again and again in a loop.



Hear Selection and Looping Options

4.2.5.1. Stop After Bar (measure)…

There is a checkbox option (***Stop after Bar***…), to specify where playing should stop. If not checked, playing will stop at the end of the piece. If it is checked, playing will stop after the specified bar (measure) or the Remembered Chord (see section 22.51) has been played. If this option is chosen, enter the required bar number (starting at 1). If there is a remembered chord, this can be specified by entering **=** (*equals*) rather than the bar number. Stopping after current selection can be specified by entering **.** (*period*).

If the checkbox is fully checked, the specified stopping place will be remembered, if the next hearing is for the same piece. If partially checked, it will only apply to this hearing.

4.2.5.2. Begin At Bar (measure)…

There is a checkbox option (***Begin at Bar***)) to specify where playing should begin. If not checked, hearing will be from the current selection. If it is checked, playing will start at the specified bar (measure) or the Remembered Chord (see section 22.51). If this option is chosen, enter the required bar number. If there is a remembered chord, this can be specified by entering **=** (*equals*) rather than the bar number.

Bar number 0 means the start of the piece (begin at the beginning). If the Hear menu is invoked with the SHIFT key down, the check-box will be will be set by default with Bar 0.

If the checkbox is fully checked, the specified starting place will be remembered, if the next hearing is for the same piece. If partially checked, it will only apply to this hearing.

4.2.5.3. *Looping…*

There is a checkbox option (***Loop***↶), that will play the piece from the selected starting place again and again in a loop until cancelled (e.g. with the ESC). The option also allows one to specify the number of times it will loop. An infinite loop count can be specified by entering ***** (*asterisk*) rather than a number; this is the default.

If both ***Loop***↶ and ***Stop After Bar***… options are active, the loop will be until the '*stop after*' place.

If the checkbox is fully checked, the looping will be remembered if the next hearing is for the same piece. If partially checked, it will only apply to this hearing.

When specifying looping there is a small checkbox option to specify whether to pause between each loop. If it is partially checked (the default), there will be a wee, silent pause. If it is fully checked, the pause will be length of a measure, and if the metronome is active, there will be a lead-in. If it is not checked, playing will be continuous.

4.2.5.4. *Start At and Stop After Special Values*

Special values can be entered for the '*Start At Bar*' and '*Stop After Bar*' values. As well as allowing specification of the remembered chord, this can make it easier to select the required measures without needing to know the bar numbers.

Code	Start At Bar meaning	Stop After Bar meaning
= *(equals)*	Start <u>at</u> Remembered chord	Stop <u>after</u> Remembered chord.
. *(period)*	Start <u>at</u> Selected note's chord (default)	Stop <u>after</u> Selected note's chord.
<=	Start at beginning of measure containing Remembered chord.	Stop at beginning of measure containing Remembered chord.
>=	Start at beginning of next measure, after that containing Remembered chord.	Stop at end of measure, containing Remembered chord.
<.	Start at beginning of measure containing Selected note.	Stop at beginning of measure containing Selected note.
>.	Start at beginning of next measure, after that containing Selected note.	Stop at end of measure, containing Selected note.
0 *(zero)*	Start at beginning of piece.	Stop if beginning of piece is reached as a result of a repeat (*unusual requirement*).
(nothing)	Ignore checkbox and start at selected note	Ignore checkbox and play until end of piece

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 for Voices with No Instrument

If checked, used midi channels (except for channel 10), 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. Speak a Voice's Lyrics or Chord Symbols

This is an experimental feature, which currently only works on Windows, and there are known issues.

If '***Speak a Voice's Lyrics***' is checked, a drop-down menu appears that allows **one** voice to be chosen. If the chosen voice has lyrics, the first lyric on each note in the voice will be spoken in time with the music. Only one voice can be used and its lyrics must be classified as Lyrics! The voice must be active in the current context, but does not have to be actually playing. Similarly if '***Speak a Voice's Chords***' is checked, annos classified as chord symbols on the selected voice will be spoken.

The text-to-speech voice used is the currently selected Windows' one; this is the one specified, using the Windows' *Text to Speech* item in the *Speech Recognition* control panel. This needs to be appropriate for the text being spoken.

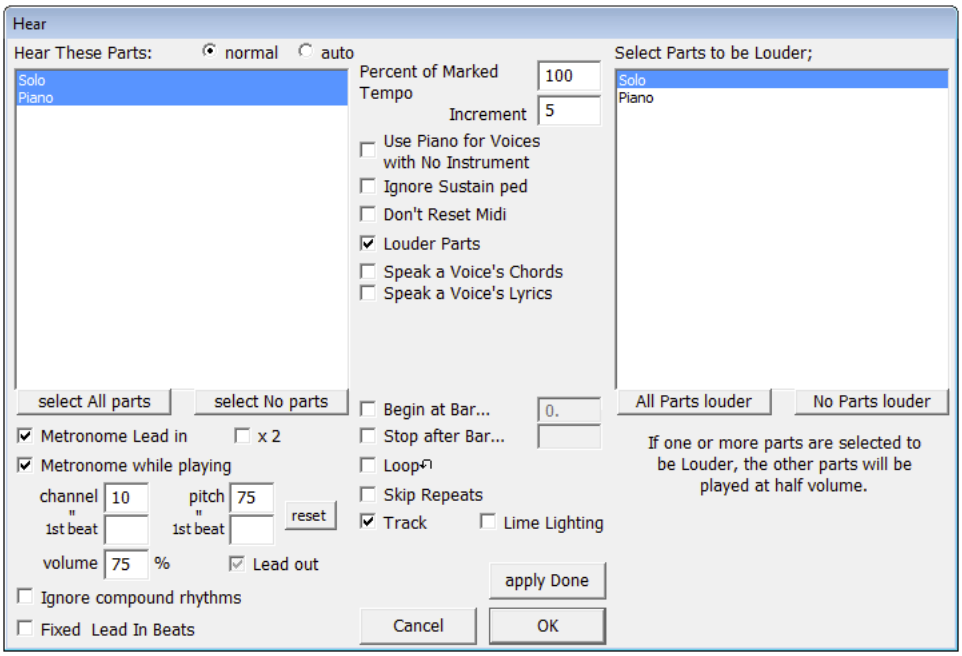
The main issue is handling lyric words that are split between notes. If a component is only 2 or 3 characters and not a recognized word, Microsoft voices have a tendency to either spell it out or treat it specially. For example, in English:

- ov – er is spoken as '`oh vee – ee are`'
- ne – ver is spoken as '`north east – ver`'
- mo – ther is spoken as '`em oh – ther`'

Further study needed to find a way of avoiding this. It doesn't seem to happen with eloquence voices.

4.2.10. Louder Parts

If checked, it is possible to specify that some parts are played louder than the rest. Typically this is used to accentuate a solo part, while still being able to hear the rest of the parts. If *Louder Parts* is selected, the dialog is expanded to allow one to specify which parts should be louder.



Louder Parts Hearing Option

4.2.11. 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.

4.2.12. No Parts Selected

If no parts are selected for hearing then, of course, nothing will be heard. A warning is shown in the dialog if this will be the case.

4.3. 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. Playing will always be to the end of the piece and, apart from repeats, there will be no looping.

4.4. 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.

4.4.1. 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 22.4.3) 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 of when done).

4.4.2. Silent Voice

If '*No Midi Output*' is set for a voice in the ***Voice on Channel*** dialog, the voice will be silent when hearing or in exported midi files, however it will still sound when the piano window (or midi input) is used or, optionally, when a note is selected. This can be useful if a voice is used to display cues within a part's context. Unless a voice uses microtones (see section 4.7), the normal setting for a voice is '*Standard Midi Pitches*'.

4.5. 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).

4.5.1. 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.

4.5.2. 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.

4.6. 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.

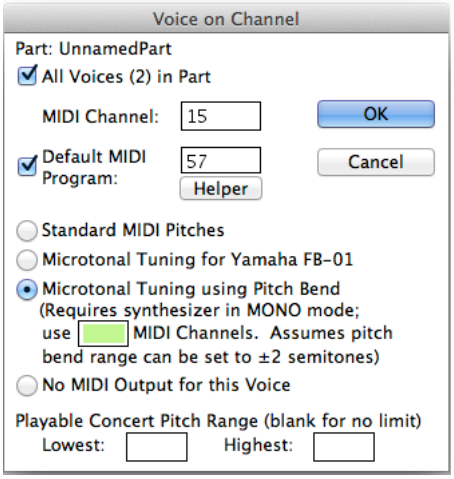
Range values are specified in '*scientific designation*[23]', with **C4** being middle C (and B3 the semitone below it). Currently the valid range is **C-1** through **G9**. If required '**#**' or '**+**' can be appended to designate sharp, and '**b**' or '**-**' to designate flat. If one of the ranges is blank (the default), then no limit will be applied for that value.

As well as the highlighting option, in 9.17 and later there is also a facility in the *Page->Fix Problems…* menu to find note with unplayable pitches.

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

4.7. 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:



Voice on Channel Dialog

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.

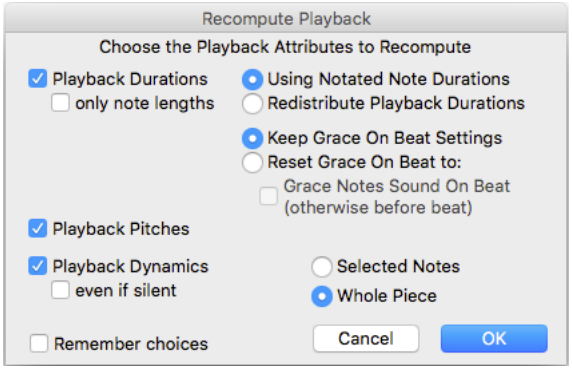
Similarly pitch bent slurs (portamentos) or ties require just one note to be playing (unless all are bent together). The maximum pitch bend is ±12 semitones (i.e. an octave).

4.8. Modify Playback

Any change made to playback using the *Hear->Modify Playback* menu will be rescinded if playback is re-computed 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 15.4.1).

4.9. Re-compute Playback



Re-compute Playback Dialog

The ***Hear->Recompute Playback…*** option allows one to reset those note attributes, which are relevant to hearing the piece. The basics are described in the main manual (c. page 94). Main additions are:

- Ability to re-compute the playback for the whole piece, rather than just selected notes.
- If dynamics are recomputed, notes that have been set to silent (zero volume) will not be recomputed unless the ***even if silent*** checkbox is checked.
- When durations are computed, there is an option to reset how **grace** notes before a chord are sounded. If chosen, the options are:

Not checked	Play note before following chord, borrowing time from previous chord;
Fully checked	Play note at time of following chord, borrowing time from following chord;
Partially checked	Borrow time from both following and previous chords.

- When re-computing durations, if the checkbox ***only note lengths*** is checked, only the sounding length of the notes is changed; the time they are played (and any grace note settings) remains unchanged. This can be useful if the default playback lengths are changed.

4.10. Basic Drum Mapping

If a voice's Clef is a percussion clef, it is possible to specify the actual MIDI pitches that are used for that voice's notes. This can be useful for hearing basic drum notation, as it would be played (MIDI channel 10 is required). See section 22.6.2 for more details.

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 means Command on Macs and Control on Windows):

cmd/ctrl-**.**	Invoke Hear option or suspend hearing (period).
cmd/ctrl-**,**	Invoke annotation placement and justification dialog (comma).
cmd/ctrl-**;**	If there is no open piece at all, invokes the *New Piece* dialog; otherwise Toggles between previously viewed context and the current one. With shift key, goes to Score context or, if already there, to last context in list.
cmd/ctrl-**'**	Toggles special accidental piano window mode (single quote).
cmd/ctrl-**=**	Remember the currently selected voice chord (equals), see § 22.51.
ctrl-alt-**=**	Remember the currently selected annotations (equals), see § 3.25;
cmd-*ctrl*-**=**	(on Macs, cmd-*ctrl*-**=** must be used, as cmd-*alt*-**=** is used by Apple).
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 dialog defaults to 'Go to Bar', but may be used to 'Go to Page'.
cmd/ctrl-shift-G	Go to 1st bar of current page. G is whatever letter (if any!) is configured as the '*Go to Bar*' short cut.

Note that on Macs, *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 built-in one remains available.

On both Windows and Macs, the set of short cuts, **ctrl/cmd-alt-letter** may be configured by the user (*and is the default for new short-cuts*). Lime will never pre-assign any of these. On Windows, the **Alt** key must be used; the alt-graphics (AltGr) key does not work for this purpose.

On Macs, *ctrl-letter* short-cuts have been replaced by *cmd-alt-letter* (***ctrl-letter is deprecated***).

5.4. TAB, Arrow Keys and Number Pad Arrow Keys 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 (that is not locked in 'Num-Lock' mode), a number of short cuts are available (*Ctl* means the Command key on Macs 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 (up-arrow or num pad 8)

by itself	go to previous voice and, optionally, play note.
+ctl	play all notes in current time-slice.
+ctl+alt	if there are annotations selected, shove them upwards; otherwise as per by itself.

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 (left-arrow or num pad 4)

by itself	go to previous note and, optionally, play note.
+alt	go to previous annotation.
+shift+alt	group select previous annotation of similar classification.
+ctl	go to previous chord and play all notes on staff.
+shift	group select previous note.
+shift+ctl	group select next voice-chord.
+ctl+alt	if there are annotations selected, shove them leftwards; otherwise as per by itself.

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 (right-arrow or num pad 6)

by itself	go to next note and, optionally, play note.
+alt	go to next annotation.
+shift+alt	group select previous annotation of similar classification.
+ctl	go to next chord and play all notes on staff.
+shift	group select next note.
+shift+ctl	group select next voice-chord.
+ctl+alt	if there are annotations selected, shove them rightwards;

otherwise as per by itself.

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.
+ctrl	take-repeat and highlight measure at its beginning.

DOWNARROW_CHAR (down-arrow or num pad 2)

by itself	go to next voice and play note.
+ctl	play from current selection to end of piece with (+shift without) metronome.
+ctl+alt	if there are annotations selected, shove them downwards; otherwise as per by itself.

5.5. Playing (and Summarizing) Selected Note or Chord

Apart from the mapping of computer keyboard letters to piano keys, as described in the old manual (Appendix A, page 195), the following will play the selected note or chord (if just 1 note is selected):

, (comma)	play the selected note;
. (period)	play all notes in selected note's staff chord;
/ (slash)	play all notes that begin in selected note's time-slice;
(space)	play all notes in selected note's voice chord.

If the shift key is used with comma or period (only comma or period), then a summary of the selection will be shown on the console, which will be made visible if possible[24] (see section 22.50 for more information on use of the Lime console).

[24] Use *Window->Show Lime Console* with shift-key to force it to be visible.

6. Reading Older 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.

Lime 9.17 and later uses unicode for text wherever possible. When reading older file, it needs to convert non-unicode text. If the score emanates from Windows, Lime may need to know the locale code page. If non-English text gets garbled when upgrading an older score to 9.17, see section 3.2.1.

Apart from that, if a score was created with Lime 9.15 or later, there should be no issues opening it in a later version of Lime. There are, however, some minor issues with some very old scores.

6.1. Upgrading Issues with Pre 9.15 Scores

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. If the result if obviously peculiar, try specifying a different "*Known Source*".

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 introduced a slightly enhanced file format[25]. 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 9.17/9.18 files but there may be some information loss (particularly with 9.15).

See section 3.2.1 for a discussion on issues with title, etc, in files created on Windows with non-English locale code pages.

6.2.1. 9.17 and Later Features

In Lime 9.17 and later, use of unicode is the recommended default and, when reading pre-9.17 files, annotations will be converted to unicode[26].

In addition to Unicode support, the 9.17 file format fully supports:

- Annotations anchored to the top or bottom of a system (see § 3.3.2.2).
- Annotation positions scalable to staff/system separation (see § 3.3.2.3)
- Vertically growable line/curve annotations (see § 3.3.2.4).
- Note by note fingering (see § 3.9).
- Auto-generated annotations (see § 3.10).
- 8va and 8vb 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** (**S**tandard **Mu**sic **F**ont **L**ayout) font identifier (see § 7.1.12).

Lime 9.19.1 and later will have a slightly newer format (***357***). Lime 9.17 and later will recognize it; however earlier versions of Lime will not.

6.2.2. Reading Older Files

On Macs or on Windows, if locale code page is essentially English, Lime 9.17 and later has no issues reading 9.15 or later scores. It is, therefore, recommended that warnings when reading slightly older files are turned off. This can either be done in the general preferences or there is a check-box in the warning message. If a file is opened within Lime using *File->Open…*, if the Shift key is pressed when selecting the file, older file warnings will be forced on for the selected file, which allows its locale code page to be specified.

On Windows in a foreign locale, you should set a default locale code page. If all your files are from that locale, then you can safely turn off warnings when reading slightly older files. However if your files come from different sources, until they have all been converted, the warning should probably be retained.

In all cases, it is recommended that you update your files to the latest format. The *Batch…* facility (*see section 22.49*) is useful for that.

---

[25] Lime 9.17 (and 9.18) use file format version **356**.

[26] Nb. all 9.16 and later versions recognize unicode annotations (actually all versions after 9.15.8)

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 tested at the time of writing, though 4.0 has been defined); 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 in complex piano parts*.

Main features not supported by LIME include:

- <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 won't import properly and will have timing errors.

7.1.1. MusicXML File Name Extensions

Files ending in **.musicxml and .xml** are treated by Lime as MusicXML files. As of Lime 9.19.5, **.mxl** (compressed MusicXML) files are also supported[27].

7.1.2. 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.

7.1.3. 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:

- Many music scanning programs create MusicXML timing errors if there were scanning problems.
- 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.

[27] It's worth noting that **.mxl** files are actually zip files and can be expanded using a zip tool (change extension to .zip).

- 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 8va and 8vb 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.

7.1.4. Timing Errors and Divisional Faults

Bad MusicXML can result in timing errors, which Lime cannot correct automatically. If any are identified, they will need to be manually fixed (see section 8).

If Lime reports a 'Divisional Error', it means there is a fundamental error in the MusicXML (missing some all important <*divisions*> elements). Lime may have corrected it, but if it hasn't succeeded, there could be strange measure lengths, even if there are no obvious timing inconsistencies.

7.1.5. Notation Ornaments

If the MusicXML specifies multiple ornaments 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.

7.1.6. Fingering and String Specification

MusicXML allows fingering to be specified for each note. Lime treats well defined fingering as a note characteristic, from which it creates an auto-generated annotation (see section 3.10) 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.

7.1.7. 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.

7.1.8. Tuplets

Lime currently 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.

7.1.9. Slurring

Lime does not support implicit slurs between staves. This means that, in complex multi-voice (e.g. 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.

7.1.10. Tempo Marking

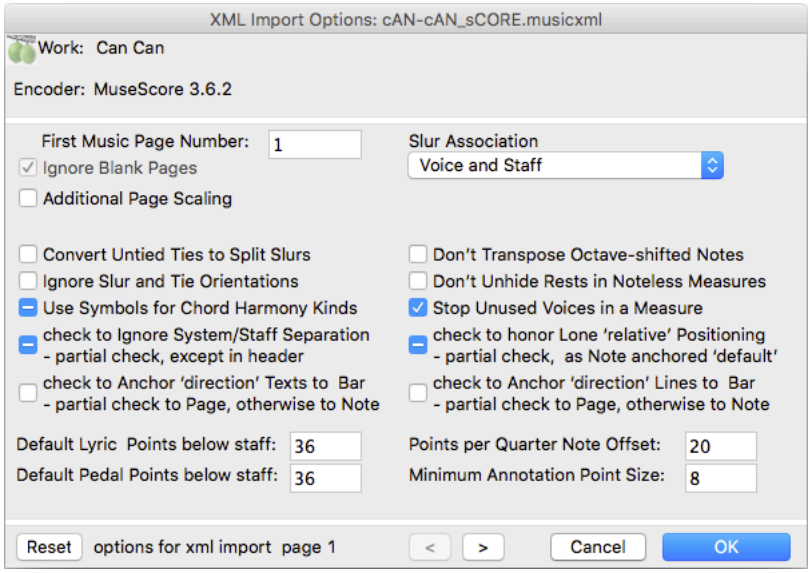
The MusicXML *<metronome>* elements are imported as tempo marking annotations. In a multi-part piece, these annotations are preset to *Duplicate in Every Part context* (see section 3.19.1). This means that, after importing, if contexts are created with just some parts, the tempo marks will still be printed.

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

7.1.11. 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.



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.
- 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.

  If not checked, 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 8va or 8vb 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

  Specifies how MusicXML slurs are associated if a part contains multiple voices. The MusicXML is ambiguous in that it defines the start of a slur (with a <slur> element), but does not identify whether the slur just applies to the voice (and staff) or any voice. Lime's default is to associate any slur to both the voice and staff. However this can be changed if the result does not seem sensible. The options are:

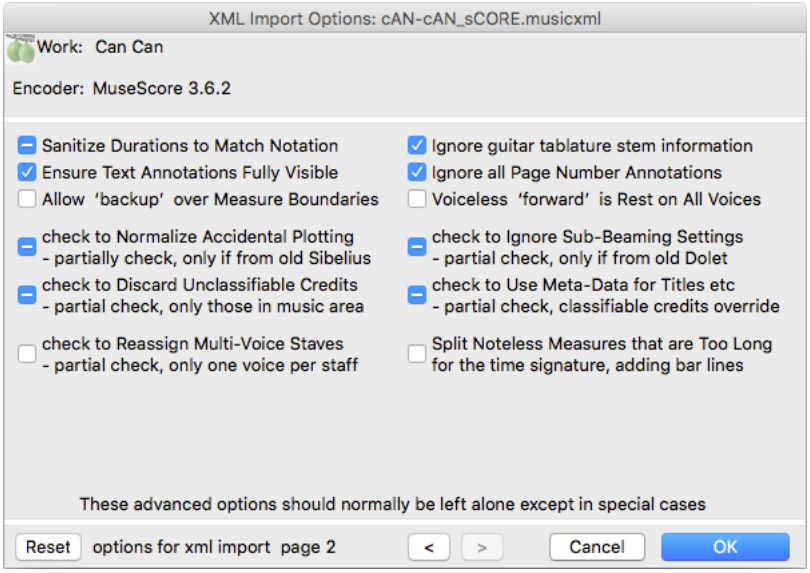
Unassociated	applies to all voices on any staff (not recommended);
Staff	applies to all notes on its starting staff;
Voice	applies just to its starting voice;
Voice and Staff	applies just to its starting voice and starting staff (default);
Voice or Staff	applies to all notes on starting staff, but if the staff doesn't match, will still apply if the voice matches; n.b. slurring between staves is not yet supported by Lime – they have to be split.

The right-arrow **>** 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.



MusicXML Import Options Advanced Options

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

7.1.12. SMuFL (**S**tandard **Mu**sic **F**ont **L**ayout)

Provided a SMuFL compatible font is installed, Lime can import annotations specified as SMuFL. The recommended font is '***BravuraText***', which is included in the Lime release package. After 9.17 any annotation can be set to use SMuFL fonts; before then, because of the file format implications, SMuFL is only supported for importing MusicXML. N.b. if you require SMuFL, you must explicitly install the '***BravuraText***' font. SMuFL is also useful if you want any music symbol that is not in Lime's fonts.

7.2. MusicXML Exporting

Although Lime can export MusicXML, the result is currently relatively primitive and is limited to basic information. This may 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. 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**.

For lyrics and other text annotations, NIFF uses single byte characters and does not specify the code page. Chances are that the NIFF file was created on Windows and uses ANSI (Windows 1252) encoding. When importing a NIFF file, it is possible to specify a different encoding.

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.

7.3.1. 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 modern music scanning engines[28].

NIFF occasionally gives better results when importing scanned files from SharpEye, for example 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. Another case is when a **note** has hairpins going to and from it. However there are very many cases when MusicXML is better. Therefore only try NIFF, if importing using MusicXML has problems.

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

7.4. NIFF Export

Support for NIFF export has been discontinued[29].

7.5. 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.

---

[28] Including SmartScore (www.musitek.com)

[29] NIFF is deprecated and, in any case, it is far more important to support exporting MusicXML.

8. Problem Fixing

8.1. Duration Errors

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 in a system do not align and the context is not polymetric[30].
- 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 though it.

The main tools for fixing duration errors are:

Fixer Context	It is possible to set up a special *fixer* context to facilitate duration error correction. A fixer context is designed to make it easier to identify the structure of the piece and where duration errors are. It can also be useful as a quick way of seeing what each voice is actually doing in a complex piece. When first set up, all voices are configured to be printed on separate staves and, furthermore, all rests are shown, hidden or otherwise. 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).
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 *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.

8.2. Fixer Contexts

A *Fixer Context* is a powerful facility for correcting problems. It is a temporary context designed to make it much easier to identify where duration and structure 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 context.

[30] 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.

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.

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:

- 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.

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

8.2.1. Voices in a Fixer Context

Fixer contexts always include <u>all</u> 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. In a fixer context, a voice can be stopped, even if it is the only one on its staff.

8.2.2. 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…*

Note that the system page (portal) number displayed in the window title of a ticker-tape view, identifies one of the systems shown; this is usually the middle system, but can be either of the others.

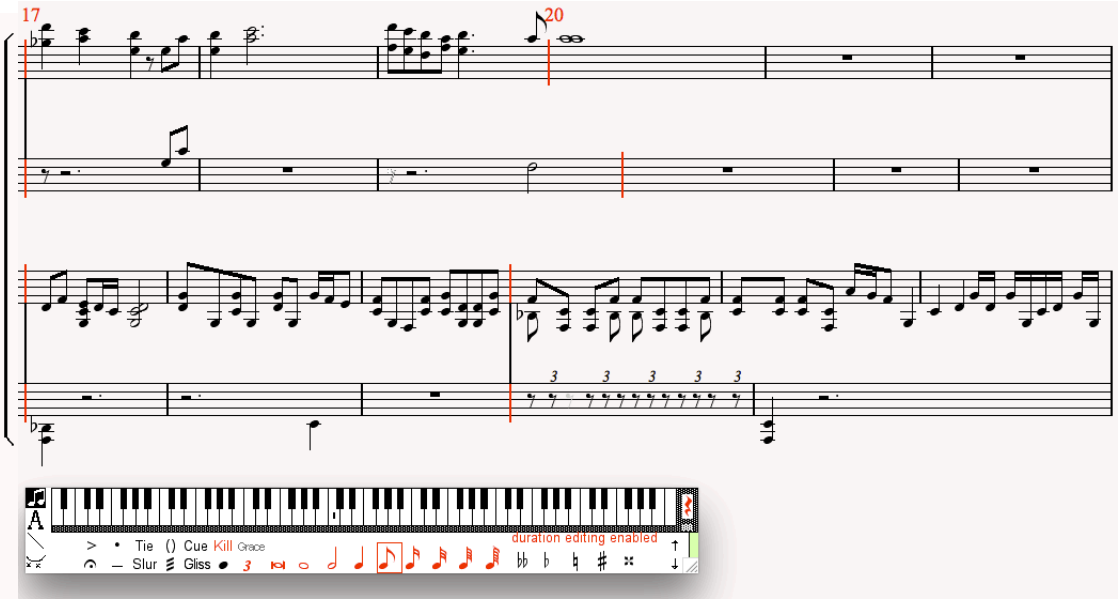
8.2.3. Oddities

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 of if voices moved between staves. Such oddities can include strange beaming or slurring and incorrect alignment of annotations.

8.2.4. 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.

8.3. 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.

8.4. 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.

8.5. Pitch Violation

Using the *Voice on Channel* dialog, it is possible to specify the highest and lowest pitch that is playable by the instrument, see section 4.6. As well as the preference to highlight such notes, there is a facility under the *Page->Fix Problems…* menu to find the next note that violates its voice's pitch limitations. This can be useful, if you are arranging or even composing music and want to be sure that the instruments you have specified can play their parts.

The *Next Pitch Violation…* menu option, will search forwards from the selected note for any note that violates the pitch limitations specified in its *Voice on Channel* (if any). If a rogue note is found, Lime will go to its page and select the note. The search is on a system by system basis, starting at the selected note. If the shift key is active when the menu item is selected, the selected chord will be included in the search, otherwise the search will start at the next note. At the end of each system, all other voices are searched. If nothing is found on a system, the next system is searched, and so on until the end of the piece. If more than one voice on a system has rogue notes, the search will cycle through them and, if to don't correct them, you will need to manually move to the next system/page.

If you are not blind, it is recommended that if the *Next Pitch Violation…* facility is used, highlighting should also be active. This is because it is easier to see what the rogue notes are. If you are a Lime Aloud user, you will be told that a rogue note has been identified, before its details are spoken.

8.6. Deleting Idle, Non-Printing Voices

If, when a context is deleted or a voice is stopped using *Parts and Voices*, a voice becomes completely idle, Lime will usually delete it automatically. However, there are cases where detection is difficult. If you believe that an unnecessary voice remains in the piece, the menu item:

Page>Fix Problems-> Delete Idle Non–Printing Voices…

can be used to identify such voices and, optionally, delete them.

A voice is considered idle if it has only rests and does not print anywhere in any context. A voice will not be deleted if it is the only voice in a part – the part has to be explicitly deleted from the score.

8.7. Fix Missing Reminders

This invokes the *Page->Reminder Clefs and Keys* dialog (see section 16.3.2), defaulting to all contexts. If the go-ahead is given, the user is told about whether any fixes were needed. Reminders can be missing in imported pieces (e.g. NIFF) and, of course, if they have been deliberately removed or hidden.

8.8. Clear Spurious Context Data…

Under some circumstances, if contexts were deleted, it is possible for there to be annotations that are not visible in any remaining context. This does not matter, of course, unless you expect to see them. However, if you believe that there could be any such spurious annotations, the menu item:

Page>Fix Problems->Clear Spurious Context Data…

can be used to identify such annotations and, optionally, clear them.

8.9. Fix End of System Bar Lines… (and bar Numbering Inconsistencies)

There was a problem prior to 9.17.6 that, if a new context without asynchronous system breaks was created based on a context, which had asynchronous system breaks, some voices could have bar lines misplaced after the end of some systems, rather than before. This could cause bar numbering inconsistencies in the new contexts. This should now be fixed, but if an existing files has misplaced bar line or bar numbering inconsistencies, the *Fix End of System Bar Lines…* facility can be used to correct them.

If you actually want no visible bar line at the end of a system, but a visible one at the beginning of the next system, do not delete the one at the end of the system, but rather make it an invisible one.

8.10. Check Page Count…

This shows the number of pages in the current context and checks that this matches number of pages being used for navigation purposes. It allows unsighted users to get a page count easily.

It has been known that, after a complex sequence of operations involving measure insertion, systems repagination and undo/redo, the number of pages shown in the music window title is wrong. This should no longer happen, but just in case it does, the *Check Page Count…* facility will recalculate the number of pages and, if there is an inconsistency, inform the user, with the option to fix it.

8.11. Corrupt or Faulty Lime File

If a Lime music file has somehow been corrupted or if there is an I/O error while reading it, the file is most likely useless. However, in extremis, as the data for each voice follow each other, it is sometimes possible to recover at least part of it. If an I/O error happens while the voice data are being read or if the file appears to be truncated, the user is given the option try to recover something. The result, of course, will be incomplete, but might, conceivably, be useful.

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 8[31], 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 two[32] 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.

Standard Colours

[31] 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).

[32] Prior to 9.16, there was only the *piece options pallete*.

9.2.1. Customisation

In all colour dialogs, it is possible to customise the 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.

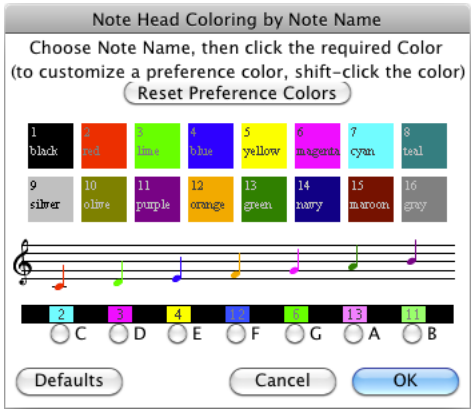
Note, however, that when the piece is printed, the result will be black on white, with normal colouring.

9.3. Colouring Note Heads and Keys by Note Name

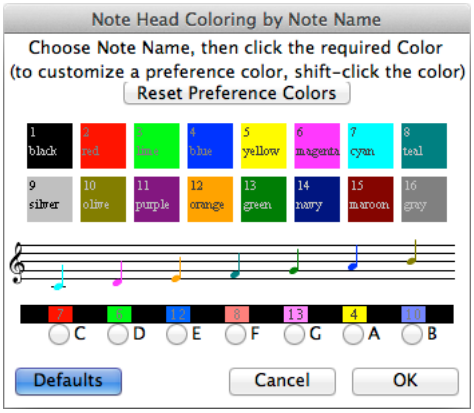
Using the preferences, it is possible to specify that every note and key signature throughout every piece is displayed in a colour based on the note name. In the preferences dialog, if the "*Color Note Heads and Keys by Note Name*" check-box is checked, it means:

- **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

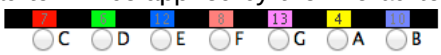
Note Head Colouring Dialog

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***	*inverted cyan*
D	**3**	**Lime** green	***6***	*inverted magenta*
E	**4**	**Blue**	***12***	*inverted orange*
F	**12**	**Orange**	***8***	*inverted teal*
G	**6**	**Magenta**	***13***	*inverted green*
A	**13**	**Green**	***4***	*inverted blue*
B	**11**	**Purple**	***10***	*inverted olive*

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 an overhaul would be desirable. 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 always override anything set in the layout option. The may be useful when precise placement is needed, independent of the layout defaults. However it is recommended that the Additional Separation parameters are used, not the absolute ones.

In Lime 9.17.8 and later, the Layout dialog has a checkbox option to clear any absolute layout parameters. If this is fully checked, all such parameters will be removed from the selected contexts. If it is partially checked, only those at the very beginning of the piece will be removed.

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 is not recommended except in exceptional cases (though it is recognized that it can be a quick way of adjusting staff and system spacing on a page).

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

In the '*Parameters*' dialog, '***Apply Change***' must be used if any change should become active.

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 for the music font in use. This can be useful, if one wants to change a parameter back to the default. The '*Default 2*' button will set the fields for the other music font.

11.2. Number Values

Number 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, better, as its numeric encoding in the specified font. Encodings are identified by there being more than one character, which is recognizable as a number. 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 music fonts (Marl, Tufa or Sonata) or the Symbol Font, the value is minus the 8-bit coding. For other 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. See Appendix B. (*Music Font Character Mapping*) for music font mapping tables.

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.

In one or two cases, such as the super-/sub- scripting values, only the Y-offset and the font size is relevant; the actual character is ignored and only used to show the specified size.

11.4. Normal Size and Cue/Grace

Many parameters are actually two parameters:

Normal Size is the value/character that applies to normal notes.

Cue/Grace is the value/character that applies to small notes (grace notes or cue notes).

These are effectively separate parameters.

11.5. 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.

11.6. 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.

The ***Minimum Default Line Width*** set for a context (defined in the layout) specifies a minimum for any line width parameter, if it is not explicitly set. If a line width parameter is specified, its value overrides the default and is, therefore, unaffected by any minimum default value.

The minimum line width for annotations is defined by the ***minimum annotation line width*** parameter (under *Dynamic Markings, Hairpins, Lines*). This will impose a minimum for line and curve annotations (overriding their line width), but does not affect anything else. Its default value is, however, subject to the *Minimum Default Line Width*.

11.7. Parameter Notes

11.7.1. 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.

11.7.2. Whole and Half Rest Characters

By default, the characters used for whole and half rests depend on whether the rest is on a staff line or not. If it is not on a staff line, it should include a wee ledger line. If, however, the character is set using a parameter, that character will always be used, whether or not it is on a staff line.

11.7.3. Page Number (Footer) Position

Since 9.17.9, the position of the 'footer' page number can be specified on the left or right and on the top or bottom of the page.

If the '*Page Number Y-Offset*' of the bottom of the page is negative, then it is with reference to the top of the page, rather than the bottom.

The new '*Page Number X-Offset*', specifies the horizontal position. If it is zero (0), then it is centred as it always has been. If it is non-zero, if it is positive, it is the distance from the left of the page and if it is negative, it is the distance from the right of the page. Furthermore, there are ***odd*** and ***even*** page options. If both are specified, then there will be different horizontal placements on odd and even Lime pages.

11.7.4. Shared Note Head Discrepancy

In 9.18.3 and later, the *Shared Note Head Discrepancy* parameter[33] has been extended to enable control of whether note heads in a chord are shared by default, so it depends on the nominal length of the notes as well as the number of dots the notes have.

Prior to 9.18.3, if there were two notes in a stem chord with the same pitch but different lengths, neither of which was whole note or longer, by default if the number of dots was the same, the note heads would be shared and the result would look like the shorter of the two, unless one was explicitly specified as "*Don't Share Note head*". For backwards compatibility, this remains the default, but can now be changed with the parameter.

The *Shared Note Head Discrepancy* parameter is now multi-value:

1 (or any odd value) means that note heads can be shared, even if the number of dots is different; this applies whether or not the notes have the same stem direction. This has always been available (even though it is unusual to do so).

2 means that, <u>if the notes have the same stem direction</u>, the note heads will always be separated if they are different lengths; the result will be two notes side by side.

3 means note head of different lengths and different number of dots can be shared.

Note that, if the stem directions are different, providing both notes are quarter tones (crotchets) or shorter, there length is obvious from the flags on the stem, so there is no need to separate them.

11.7.5. Bar Line Left of Single-Staff System

By default, a bar line is not drawn at the left of a single-staff system, unless there is also a bracket. The *Draw Bar Line Left of Single Staff System* parameter (under *Brackets and Braces*) allows this to be changed:

0 Never draw a line;

1 Always draw a line;

-1 Draw a line provided the system is at a measure boundary (useful for polymetric systems),

-2 same as -1 but do not default to drawing a bracket (recognized in 9.18.8 and later).

11.7.6. Slurs Over System Breaks

As of 9.18.8, there is a parameter that gives some control over the appearance of slurs over system breaks. By default, slurs over the end of a system prefer to be horizontal, but those onto the beginning of a system prefer to be normal. The *Slurs Over System Breaks Like Horizontal* parameter can be used to change this:

0 No attempt is made to draw them horizontal;

1 Both ends and beginning will be horizontal, when practical;

-1 Slurs over the beginning will not be horizontal, but ones over the end will be (this is the default, as of 9.18.8);

-2 Slurs over the end will not be horizontal, but ones over the beginning will be (not recommended).

[33] In parameter class "DOTS FOR NOTE VALUES" and in class "NOTE HEADS".

11.7.7. Ties Like Opposite Bowing to Slurs

As of 9.18.8, there is a parameter to set the default direction of a tie curve if the tied notes are also slurred.

- 0 Ignore any slurring (default);
- 1 Always prefer opposite direction to the slur;
- -1 Prefer opposite direction to slur, but not if the chord has notes in both stem directions.

11.7.8. Tied Chords Like Only One Tie Curve Drawn

As of 9.18.8, there is a parameter that will draw only 1 tie curve if a multi-note is tied.

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. 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 piece's editing zoom.

12.2. Temporarily Zoom to Fit Screen

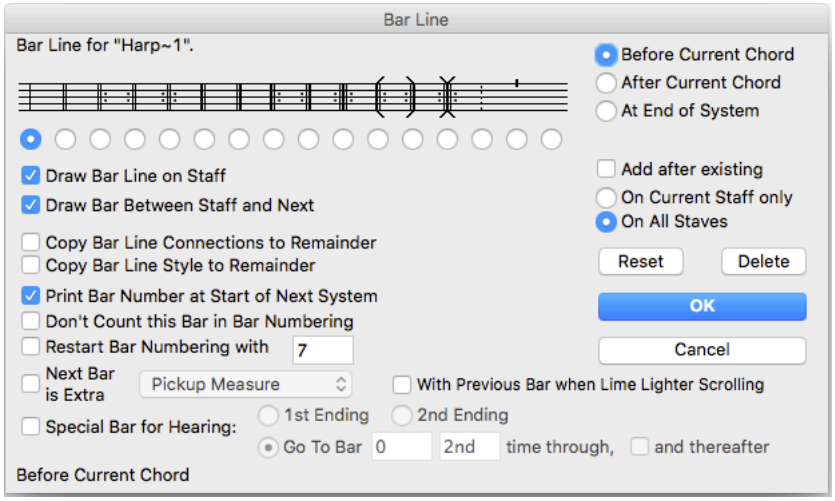
If the music window's maximize button is clicked with the *Alt* key pressed, the music window will be temporarily enlarged and zoomed to fit the screen width. Clicking it again (without the *Alt* key) will revert to its previous zoom (on Windows, the Frame maximize button can be used to revert).

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. Bar lines are always in all contexts.

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 chord, after it, or at the end of system. The dialog should then display the characteristics of any existing bar line at that point. Note that a bar line after the last chord in a system is the bar line at end of system; so if you want a multiple selection to include end of system, choose last chord in each measure.

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 fully described in the old manual.

13.1. Restart Bar Numbering

Restarting bar number should only be used at a section boundary, where you really want to restart the numbering (usually from 1 or 0 - and usually such restarting should also be on a page boundary). Normally, '***Don't Count the Bar in Bar Numbering***' is sufficient to cater for pick-up measures, etc.

13.2. Hidden Bar Lines

If a bar line is drawn neither on staff nor between staves then it is invisible in this context. However, if you really 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 virtually <u>no space</u> in any context.

13.3. 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 with the next measure rather than highlighted by itself.

13.4. 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. repeat and then *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.

13.5. Contexts

If not in a context using all parts, take care if inserting or deleting bar lines or changing numbering.

14. Slurs and Ties

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 tried to associate slurs to stem direction. 9.15 and later includes options for them to be more voice associated. The 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 an unslurred note whose stem is in the same direction as the slur itself. The slur will terminate at such a chord, but there are exceptions:
  For example, If a slur comes from a single stemmed chord into a multi-stemmed chord, in which the single stemmed voice is now in the opposite stem direction and is slurred, the slur will continue even if its association is not voice associated.
- **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). Currently, the default is the original stem direction algorithm, but that may change after 9.19.

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.

14.1.1. MusicXML Importing Slurs

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 forever[34]!

Unfortunately, the MusicXML facilities to define slurs for multi-voice parts are ambiguous, particularly if there is more than one voice on a staff; so after importing such pieces, it may be necessary to fiddle with the slurring so that it look and sounds correct.

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.

[34] Clearly this needs to be detected and resolved!

14.3. 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:

Prefers Note Head Specifies that they should actually be drawn so that the end is as near to the note head as possible.

Prefers Stem End Specifies that no attempt should be made to move the end closer to the note head.

These options only have any effect if it makes sense, given the stem direction of the chord and whether the slur is *On Top* or *On Bottom*. Note that even whole notes have a nominal stem direction.

As of 9.17, these may be specified on the terminating chord. *Prefers Note Head* is also available for alternate slurs; *Prefers Stem End* is available after 9.18.1[35].

14.4. Grace Slur

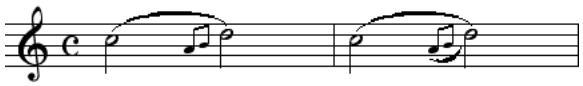
The Grace Slur sub-menu item specifies whether to draw a slur appropriate for grace notes. This is set by default for grace notes. For ordinary notes this can be useful when slurring from a regular note to a *nachschlag* (this can also be achieved with an additional slur).

If one of the stem menu items to add a slur to a group of notes (i.e. more than 1 selected) is used and if one or more of the notes are flagged as grace slurs, then the flag will be set for all notes in the group.

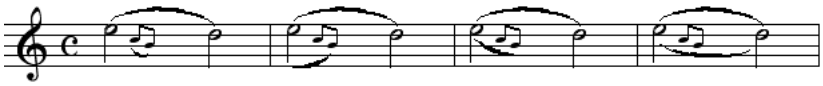
14.5. Additional Slur

Additional Slur allows one to include a second slur symbol for a note. By and large, additional slurs are treated like Grace slurs; they will attach to grace notes in preference to normal notes.

Normal slurs are drawn independently of grace notes, whether or not the grace notes are slurred, as shown in the following example:



When a regular note has *nachschlag* grace notes following it, an additional slur can be used to slur to the nachschlag, as shown in the 2nd measure in the following example (the additional slur could be further modified to prefer note head as in the 3rd & 4th measures):



Normally, if a slurred note has a nachschlag, if the last *nachschlag* is slurred, the *nachschlag* slur is not extended. This is because it is normally undesirable to have two slurs between the main notes, even if that is how it sounds. However if the final *nachschlag* has an additional slur, then the slur will be extended as in the 4th measure above.

[35] grace/alternate slurs needed an incompatible file format change to support *Near Note Head* specification.

Normally a grace note slur terminates on its associated normal note. If there is also a slurred to *nachschlag*, the slur is normally split. However, if the last of the normal grace notes has an additional slur, and the slur directions are the same, the slur will be extended over the normal note to the end of the *nachschlag* slurring as in the 2nd measure in the following example:

14.6. 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 slur[36]. 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.

14.7. Conversion of Old 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.

[36] 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 was 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 were 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 Key Signatures and Clefs

Clefs and Key Signatures 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 (and will be a copy of the previous symbol, whether hidden or not).

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

Unless reminders have been turned off for the context (see section 16.3.2), a copy of the prevailing symbol is always shown. Deleting a reminder, as suggested in the old manual, has no visual effect. If you want a reminder not to be shown, then it can be hidden using the appropriate symbol dialog.

Hidden key signatures can be quite useful (see below), but hidden clefs are very rarely wanted.

15.4.1. 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.

15.4.1.1. 8va and 8vb Passages

A hidden Key Signature is also particularly useful for temporarily setting octave-shifted play-back, such as 8va and 8vb. Place a hidden key signature before and after the required section and specify the required octave shift (see 22.5)

- *before* specify +1 for 8va, -1 for 8vb, 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.

As an octave shifted key signature look exactly the same as a normal one, there is no problem with reminders at the beginning of systems.

15.4.1.2. 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 22.4.2).

15.4.1.3. 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

15.4.2. Hidden Clefs

Partially hidden Clefs can sometimes be useful when voices move between staves to temporarily use a different clef. Normally the voice will use prevailing the clef. However, if such changes happen at the start of a system, the normal clef of the voice may be used. In that case partially hidden clefs before and after the move will be needed. In such pieces, it is recommended that partially hidden clefs for changes that happen at the start of measures, to cater for system structure changes that move them to the beginning of a systems.

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. Other than to avoid such duplication, fully hiding Clefs is not recommended, except in special cases.

Use of hidden clefs for 8va and 8vb passages is <u>not</u> recommended, particularly over system breaks because the reminders will be wrong; hidden key signatures, with octave shifted play-back work well.

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. New or Old file format Warnings

By default is a piece is opened that has an older or newer file format, the user is warned when the file is opened. This drop-down menu allows one to specify when warnings are issued

No Warnings.	No warnings for any readable file.
Any Older or Newer files.	Warnings are always issued for any different format.
Newer Files, not Older files.	Warnings are only issued for newer file formats, older files are opened with no warnings.
All Except Slightly Older files.	Warnings are not issued for the last file format as it is pretty compatible.

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

16.1.2.1. Setting Default Character Set

When reading any pre-9.17 file, if there is no preference specifying what the default character set is, a warning will always be issued, so that the character set can be specified. The little ***details*** button to the right of the drop-down menu (on Macs a little right arrow), allows one to specify the default.

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 default 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 25.8).

16.1.4. Metronome is default when Hearing

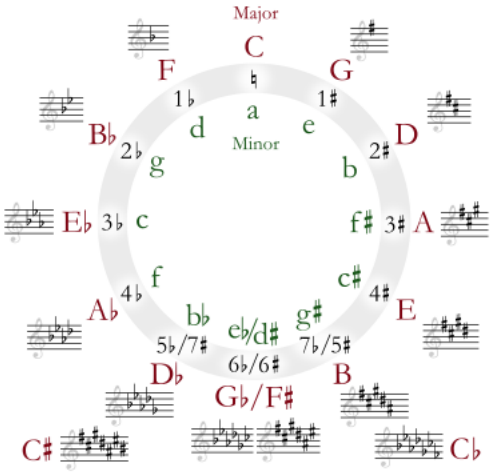
If this is checked a metronome will be the default for the session when the Hear option is used; if it is fully checked, then a lead-in will also be default.

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* preference[37] 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). In addition in 9.17, there is a facility, under the Page->Fix Problems menu, to find the next note whose pitch is outside its voice' specified range.

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.

[37] The preference to 'Identify Note with Modified Pitch' highlights (usually in Orange) notes 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.

16.1.7. 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.

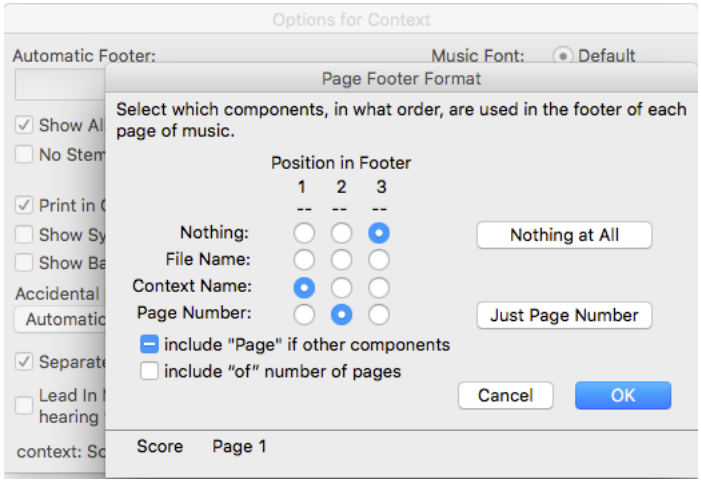
16.1.8. Reminders to Save File

If the piece has been modified and this option is set, when structural changes are likely, there will be reminders to save the piece. If the option is fully checked, there will also be a reminder whenever you change pages[38] while editing the piece.

16.2. 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 the '*include "of" number of pages*' checkbox is checked, the footer will include "*of*" the number of pages in the context.

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

[38] This is because, in the very old days on small computers, changes couldn't be undone after a page change.

16.2.2. Showing Accidentals

When music is entered, appropriate accidentals for notes are chosen. If possible, notes are represented without accidentals, either because of the key signature or because of a previous one in the voice's measure). This can be overridden using the *Note->Show Accidental* menu item.

16.2.2.1. Automatic Accidental Showing

In Lime 9.18.6 and earlier, showing of accidentals was always voice based. If, in a context, more than one voice is printing on a staff, there can be conflicts between the voices. So, after 9.18.7, Lime provides a context based option to automatically show accidentals appropriate to the staff, taking account of all voices on the staff. In the *File->Options for Context…* dialog there is a drop-down called *Accidental Showing Algorithm*, which provides one of three choices:

- **As *Specified by Voice***. This is (currently) the default, and accidentals are chosen on a voice by voice basis, like they always have been. The accidentals are set when the music is entered.
- ***Automatically***. This plots accidentals automatically based on all voices printing on each staff. This is recommended (and will soon be the default for new pieces).
- ***Automatically by Stem Direction***. This treats each stem direction separately. It is only recommended if the voices on the staff have uniform (and opposite) stem directions.

If one of the automatic options is activated for a context, the voice based *Note->Show Accidental* option is ignored, unless "**: *even if automatic***" is also checked. If showing or hiding a note's accidental and "**: *even if automatic***" Is chosen, the effect will apply to all contexts in which the voice appears. When Shift is used with *Note->Show Accidental*, it will also flip the ": *even if automatic*" setting.

Normally the ": *even if automatic*" option is used to show a reminder accidental; however it can be set to ensure an accidental is not shown, not even if automatic. Forcing an accidental not to be shown is unusual, so there are options in the *Note* menu to "*Clear Auto Accidental Hiding*" or "*Showing*". There is also a facility in the *Edit->Clear* menu to *Clear Auto Accidental Hiding or Showing* in the whole piece.

16.2.2.2. Separate Grace Accidental Showing

By default, accidentals on grace notes carry through the measure and will affect full-sized notes. This can be confusing as grace notes and their accidentals are small. So, in the *File->Options for Context…* dialog, there is a check box that allows one to specify that grace note accidental showing is semi-separated and that grace note accidentals do not always carry through.

16.3. Piece Options

In case it is of interest, the *Piece Options* dialog shows the type of system (Mac or Windows) on which the piece was last saved. If preceded by '-' or '<' it was an old file format ('<', very old); if preceded by '+' it was a newer file format.

The following sections identify new or changed piece options:

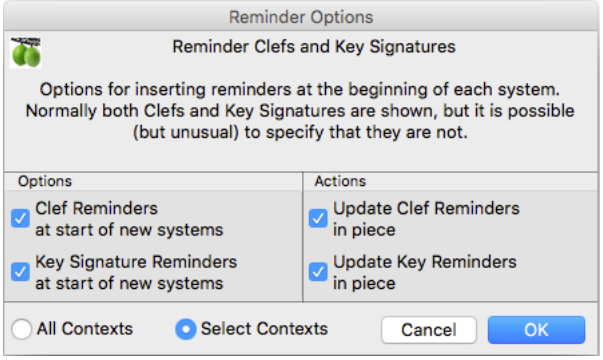
16.3.1. Uniform Marks as in 9.05 and earlier

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.

16.3.2. Reminder Clefs and Keys

The piece options for inserting reminder clef and/or key signatures at the beginning of new systems (in all contexts) have been replaced by a new Page menu option, *Page->Reminder Clefs and Keys…* This invokes a dialog that allows the options to be applied to selected contexts and, moreover, to update the piece to reflect any changes.



Reminder Options Dialog

The options check-boxes specify what reminders are shown at the beginning of new systems. If the corresponding action is checked (on by default), the setting will be applied to the whole piece to the selected or all contexts. Applying the setting will ensure that all systems have the specified reminders, adding or removing them as required.

Normally, except in specialized scores, reminder Clefs and Key Signatures are always shown on each staff at the beginning of each system.

If the *Select Contexts* radio button is clicked (even if already set), A sub-dialog is invoked to select the contexts, to which the options and actions will be applied. When invoked from the *Page* menu, the default is all contexts, on the assumption that a specialized context is selected.

If all actions and *All Contexts* are selected, reminders will be fixed throughout the whole piece. For this purpose the dialog can also be invoked from the *Page->Fix Problems* menu, in which case it defaults to all contexts.

If an action check-box is not set, the associated reminder option will be applied to any new systems that are created (e.g. using *Page->Systems*), but no changes will be made to existing systems.

17. Polymetric and Polytemporal Parts

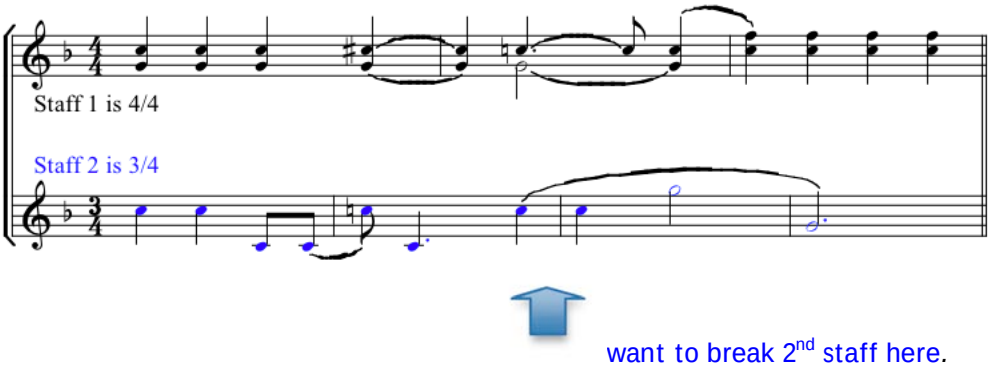
Lime 9.15.9 included support for system separation and pagination in pieces, which have parts that proceed with multiple independent beat rates and whose measures do not align because they are of different lengths. This can legitimately happen for two reasons (within Lime the term '***Polymetric'*** 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 17.5) 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. *This advanced feature should only be used if you know what you are doing*.

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 now allows a staff to be split anywhere, even in the middle of a chord. After the split, everything remains aligned.

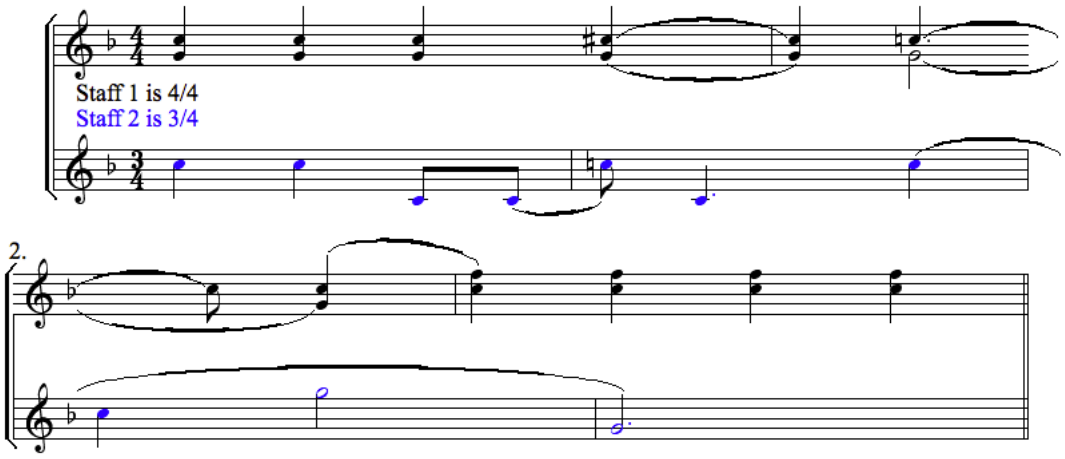
There is an option for a piece's context (*Options for Piece…* menu item) that specifies that the piece has *Polymetric Parts* (or polytemporal) and system breaks can be inserted anywhere, even if not at a timing boundary.

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.



Lime can, however, split at this point and correctly align the music afterwards. What happens is that any note, which 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) and, if the top staff has a split measure (as in this case), the measure number after the split may be followed by a comma (*9.17.2*).



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 also 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. Bar Numbering Inconsistencies

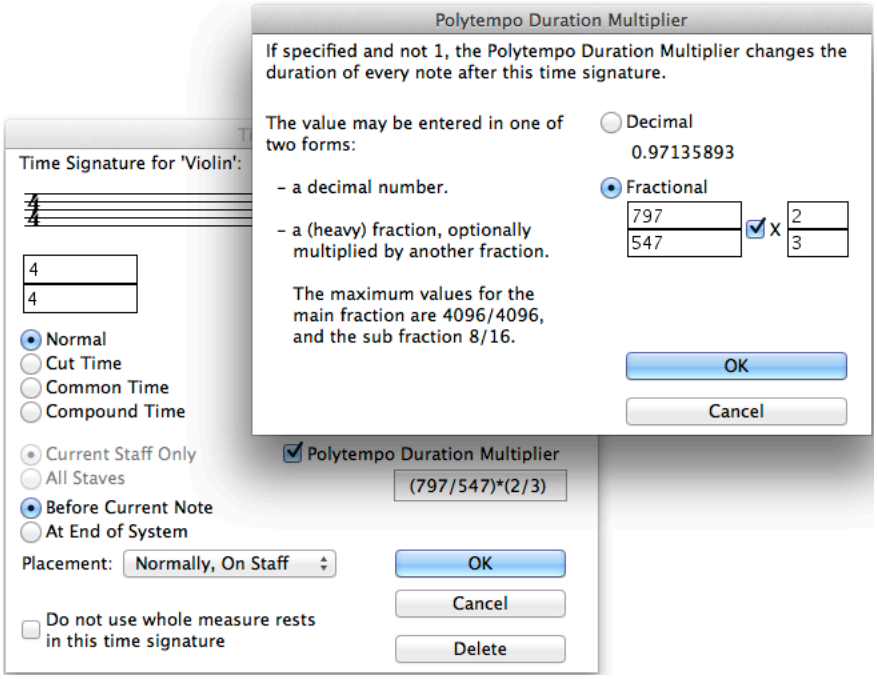
There was a problem prior to 9.17.6 that, if a new context without asynchronous system breaks was created based on a context, which had asynchronous system breaks, some voices could have bar lines misplaced after the end of some systems, rather than before. This should now be fixed, but there is a problem fixing option (see section 8.9) if existing files have numbering inconsistencies.

17.5. Polytempo Note Duration Multiplier (very advanced feature)

If a time signature has a *Polytempo Duration Multiplier*[39], 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.

[39] 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.

17.5.1. 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.

17.5.2. 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.

17.6. 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.

17.6.1. 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.

17.6.2. 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.

17.6.3. 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.

17.6.4. 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 (staff related); after 9.15.9, it is an attribute of a time signature and properly integrated.

18. Parts and Voices

18.1. Parts and Voices on staff… Dialog

18.1.1. Arrangement Propagation

One of the options in the (complex) '*Parts and Voices on staff*' 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 is 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! If a new part is created, new staves will be created automatically; this is also an option when creating a new voice.

If a new stave is created, it will be created and shown throughout the whole piece in that context, regardless of the current selection. If you want it hidden before the selection, go back to the start of the piece and set it 'not printing'; then go back to the selection and set it printing again.

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.

18.1.10. Uniform Stem Directions

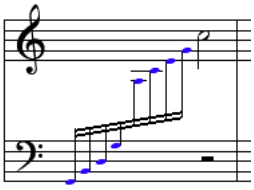
If, for a staff, the *Uniform Stem Directions* checkbox is set, each voice on the staff will be forced to its specified stem direction (if one is set), except for notes that are explicitly set to another direction. In 9.17.2 and later, if for a voice, neither stem direction is selected, its note's stem directions will not be forced. If you click a stem direction that is set, it will be cleared. If you click a stem direction that is not set, it will be set and the other direction will be cleared.

This partial *uniform stem directions* capability[40] can be useful if a staff has more than 2 voices, and incompatible note lengths in some places. A degree of uniformity can be set, but Lime may be able to adapt stem directions for a voice if there is incompatibility and thus avoid layering.

[40] Pieces created with partial *uniform stem directions* should, <u>but may not</u>, look right in Lime 9.17.1 or earlier.

18.1.11. Grace Notes

There is a recognized issue if a voice has a grace note sequence that moves from one staff to another (using *Voice to Staff Below/Above*). The *Parts and Voices on staff…* dialog applies any changes to the notes that are at the same system time on each voice. For this purpose, grace notes are considered zero length. This means that any changes made by the dialog will be before all grace notes before a chord and after any nachschlag ones. In such cases, if changes are made, you may need to reapply *Voice to Staff Below/Above* to relevant notes in the grace note sequence to get them back to the desired staff.



Grace Note Sequence Changing Staff

18.2. Voices in Parts…

The *Voices in Parts…* option, under the *Voice–>* menu, allows one to browse parts and to re-assign voices between parts. Parts may be merged to create a single part or may be split into multiple parts. The option applies to the whole piece and is available in any context. Any changes made are undoable and, moreover, should not affect the visual appearance of the piece in any context.

The primary use of the option (introduced in 9.17.2) is to rationalize parts after a (MusicXML) piece has been imported. However the capability can be generally useful, particularly in complex scores. For example, if one wishes to hide idle staves, it may be desirable to split a part so one of its staves may be hidden, or to merge two parts so one of them doesn't get hidden.

¿ *This section needs to be expanded; following is a brief summary.*

The *Voices on Parts…* option invokes a dialog. Select one or more parts, then choose the function you want to perform, then if appropriate, select the voices and enter the required part name.

- Split — Select one part (or more) and choose the name for the part that will be split off. The selected voices will be split off into a new part, with the specified name. Voices that are not selected are left on their existing parts.
- Merge — Select two (or more) parts and choose the name for the merged part. This name may be the name of one of the selected parts. If not all voices are selected, the unselected ones are left on their existing parts[41]. It should be noted that if parts that are in different contexts are merged, the new part will appear in all affected contexts, but some voices may be stopped from the beginning in some of them.
- Rename — Select one (and one only) part and specify a new name.
- Browse — Allows one to select parts and see what voices they contain. Nothing whatsoever is done of OK is chosen, when *Browse* is chosen.

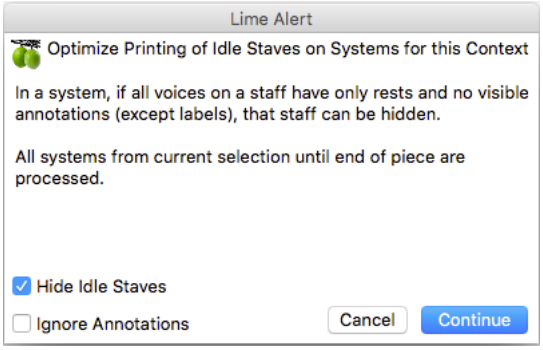
[41] Merging and splitting parts are essentially the same function. They are separated to apply different defaults and to make it a bit clearer what will be done.

18.3. 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

Currently[42] 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.

[42] If required, in future versions, it would be possible to make enhancements to specify a subset of parts to process and, also, to specify a stop point (bar) for the processing.

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 (see section 18.2).
- 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. Apart from simple changes of voices in the same part on a grand staff, if there is a change mid-system whereby a part hops to another stave or another part hops to a hidden stave, the resulting stave 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.

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*)[43]. Special accidentals can be used either in note entry mode or to modify the accidentals of an existing note.

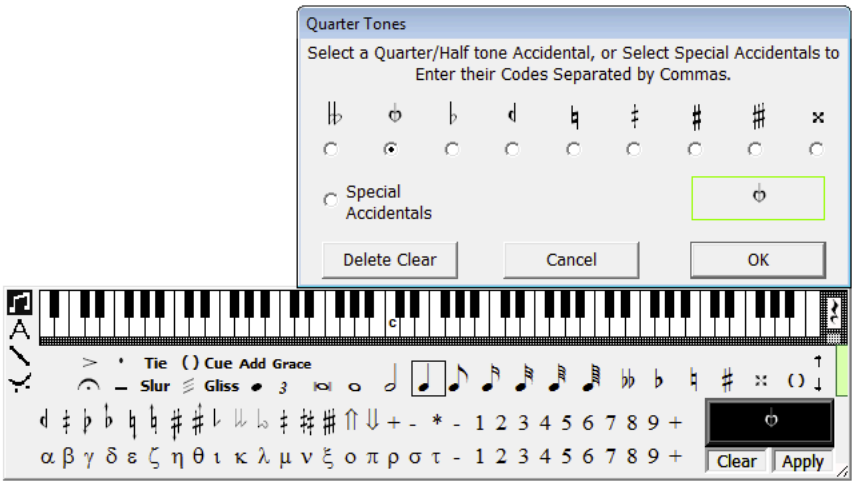
- Note entry mode — Select the required accidental(s), then play the associated note.
- Modification mode — Select required note(s) then select the required accidental(s), then click the ***Apply*** button.

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 I 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 normal.

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-*) twice very fast.

[43] The extension can also be shown or hidden using *Windows->Piano Windows Zoom->Show Special Accidentals*.

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 adjustments[44].

19.3. Hearing Special Accidentals

Unfortunately, the MIDI standard (even in 2022) 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

For pieces created with 9.17 or later, standard quarter-tone special accidentals are used. Parameters are available to change the character details (but not the pitch). N.b. currently, three-quarter flat, 𝄫 is actually a combination between a quarter flat and an ordinary flat.

However, prior to Lime 9.17 release, the quarter tone dialog relied on the pitches of the quarter tone special accidentals not having been changed, particularly for quarter flat, quarter sharp and three-quarter sharp (special numbers 1, 2, 14). It was strongly recommended therefore that if practical, the 'cents offsets' of the first 14 special accidentals were not changed.

However it was recognized that, for some uses, it was necessary to change the definitions and offsets of most of the special accidentals from 10 and upwards. In that case, it was recommended that special accidental 9 (𝄾) was changed to the 'standard' three-quarter sharp (♯, with 150 cents offset)[45]. Lime would detect this and use it instead.

[44] 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.

[45] The 'standard' 3/4 sharp is a lower case '**n**' (-110) in Tufa font, 24pt and with a y-offset of -2.

20. Ticker-Tape (panorama) and Autocue Contexts

Ticker-Tape[46] 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 like a 'teleprompter'. 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 8.2).

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 (panorama) 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.

[46] Ticker-tape and Autocue contexts 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 (pseudo) 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).

In a ticker-tape or autocue context, the left or right arrow keys will not stop at page boundaries, but will behave like TAB and automatically move to the next or previous page as appropriate.

20.3. Nominal Page Width of a Ticker-Tape Context

The nominal page width of a ticker-tape (panorama) 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.

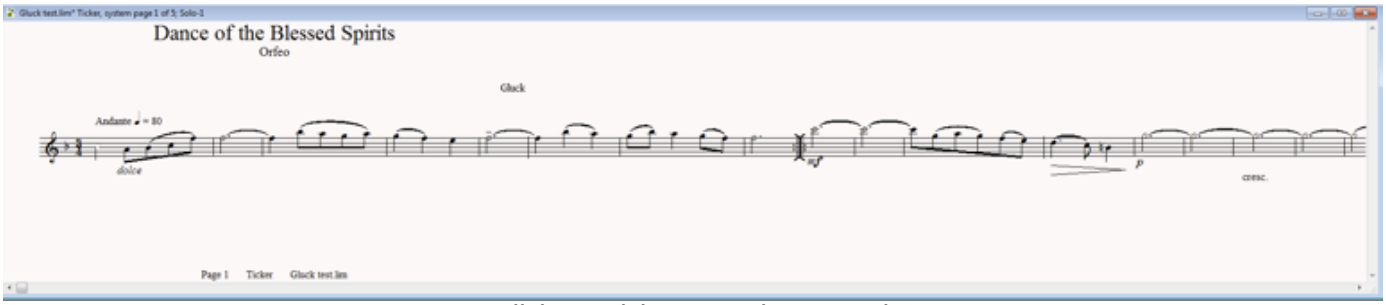
20.5. Page Size & Zooming

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 (panorama) 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.

Note that the system page (portal) number displayed in the window title of a ticker-tape view, identifies one of the systems shown; this is usually the middle system, but can be either of the others.

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. Piano Windows

The horizontal Piano window is the primary means for entering music into a Lime score and adjusting the details.

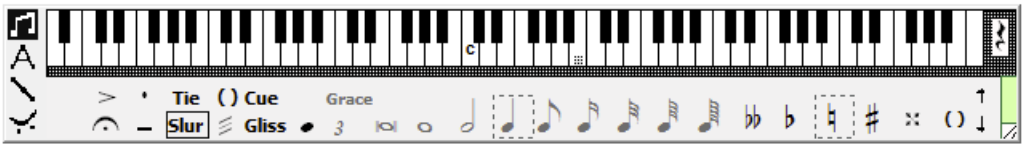
As of Lime 9.18.9, there is also a vertical Piano window (*see section 21.2*). This provides a staff-oriented method of entering notes and is designed to assist users who are not acquainted with a piano keyboard or are not familiar with the Clef into which the notes are being entered.

21.1. Horizontal 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, if the shift key is down, the window is repositioned to its default position. Other sizes may be set, by using the mouse in the grow-box at the bottom right; but this is not recommended.

The piano window may be moved to a new position using the drag box - the box on the right of the window; this is light green. Double clicking this box will move the piano to its default position.



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.

The little **c** is on the Middle C. The slightly hashed note indicates the selected note (in this case B4).

21.1.1. Piano Window Mode

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

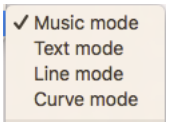
- Music mode for viewing and editing notes;
- Text mode for creating and editing text annotations;
- Line mode for creating or modifying Lines, etc;
- Curve mode for creating or modifying Curves.

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.

21.1.1.1. Setting Mode Menu Item



As of Lime 9.19.2, the music editing mode can also be changed using the menu *Annotation->Annotation Creation Mode* =

When one of the menu items is used, it will put Lime into the selected mode, prepared to create the selected type of annotation (unless Music mode was chosen). If the SHIFT key is used when invoking the menu item, it behaves exactly like clicking on the equivalent piano window icon.

21.1.2. 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.

21.1.3. Annotation Mode Status Indicators

Annotation mode (Text, Line or Curve) can be set either by clicking on one of the Piano annotation mode buttons (or using the creation mode menu item or by selecting one or more annotations. In annotation mode, a summary of the selected annotations is displayed at the bottom of the Piano.

If a text annotation is being edited, the information includes the basic font size at the start of the annotation. This can assist in understanding the sizing in backslash expressions.

If more than one type of annotation is selected, each associated mode will be highlighted.

21.1.4. 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).

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 keyboards[47], NUM-Lock is toggled using the key above the '7' on the number-pad. On Macs the SHIFT-key must be used as well[48].

---

[47] On some keyboards (notably the Dell KM714), NUM-Lock is permanently on and cannot be changed.

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

21.1.5. Piano Window Zoom menu

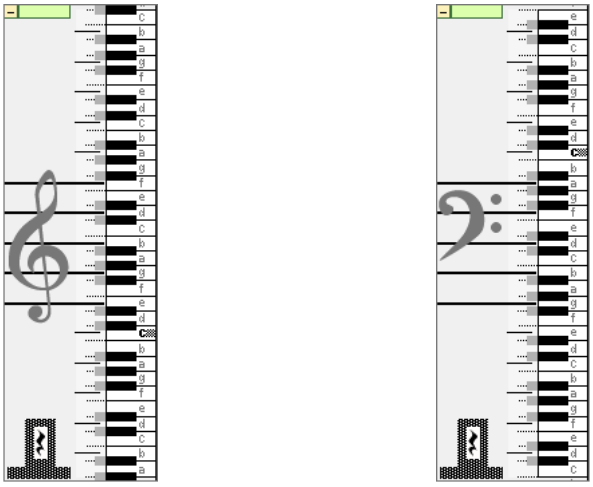
The *Window->Piano Window Zoom* sub-menu can be used to change the size of the piano window and also whether it is shown at all. In addition, it contains the option to show the vertical piano window.

The *Minimize* and *Show* items are toggles:

Minimize Piano	Toggles Minimization of the Piano window;
Show Piano, floating	Always restores it. If used with the Shift key, it is restored to its default position (on the default display) and, furthermore, if the Vertical Piano is active, restores it to its default position. Use of the shift key with any of the zoom% options, has the same effect (as well as changing their zoom).

21.2. Vertical Piano Window

As of Lime 9.18.9, there is also a **Vertical Piano** window. This can be shown using the menu item *Window->Show Vertical Piano*; it is checked if the vertical piano window is active (partially checked if it is collapsed). The Vertical Piano's visibility is remembered; so if visible when Lime is closed, the Vertical Piano will be visible next time Lime is opened.



Vertical Piano Examples

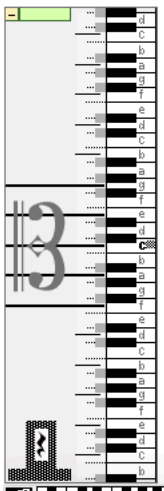
The 'keyboard' of the Vertical Piano behaves like that of the horizontal Piano. If clicked, it will play or enter notes. The actual keyboard position and octave depends on the prevailing Clef. The keyboard notes are aligned with the staff where the notes would appear. This means that one can click the 'key' that is associated with its position on the staff (and ledger lines). The naturals can also be clicked in the staff/ledger line area indicated by the dotted lines. Sharps or flats can also be clicked on their little black key extensions. A rest may be entered using the rest symbol at the bottom.

Each white note is labelled with its name. Middle C is slightly emboldened. In the examples above, the selected note is middle C.

The aim of the Vertical Piano is to provide staff-oriented note entry as though one were clicking on the actual music.

The Vertical Piano only contains the keyboard; for other data, the horizontal Piano must be used.

21.2.1. Vertical Piano Position

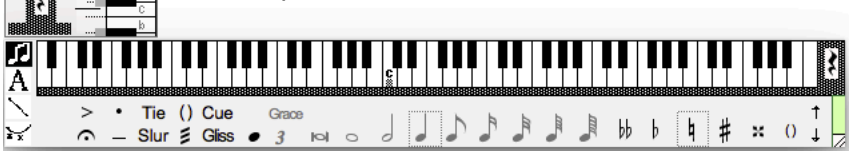


By default, when shown, the vertical Piano is positioned at the left of the horizontal Piano and just above it. Furthermore, by default, if the horizontal Piano is moved, the vertical Piano will move with it.

The vertical Piano does have a drag box (the little green one at the top) and, if it is dragged, it becomes independent of the horizontal Piano. Double click in the vertical piano's drag box changes its position to its standard position w.r.t. the horizontal Piano. If double clicked with shift key, it will be attached to the horizontal Piano.

Double click in the horizontal piano's drag box, sends the Piano window to its default position (bottom left). Clicking with shift key attaches the vertical piano (if visible).

If, for any reason, the horizontal Piano is collapsed, the vertical Piano will be hidden. The vertical Piano can be independently collapsed (minimized) by clicking its little top-left button.



22. Miscellaneous Items

These are in no particular order.

22.1. 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).

22.2. 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 you 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.

22.3. Dialogs

22.3.1. Entering Numbers in Dialogs

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

22.3.2. Tri-State Check-boxes

In many 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 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.

22.4. Note Entry

22.4.1. 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.

22.4.1.1. Nachschlag Grace Notes After a Note

To enter Nachschlag grace notes double click required note to enter note entry mode; double click ***Grace*** in the piano window and, in the wee dialog, choose '*After the note (Nachschlag)*' then enter the required nachschlag grace notes. Piano window will show ***Nach*** if you are in Nachschlag mode.

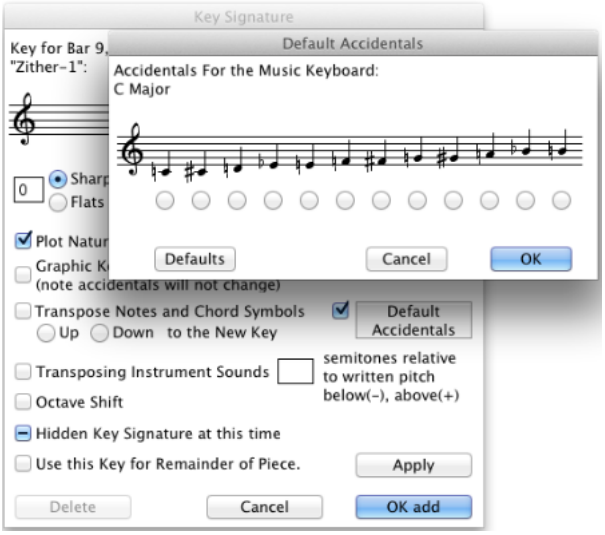
In step mode, there is special handling if the chord is specified as a Nachschlag grace note:

1. If the insertion-point is a normal note, the Nachschlag is placed after any other Nachschlags already on the note (or, 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, if there's 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 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.

22.4.2. 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.

22.4.3. 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 22.5.3).

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 22.6).

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.

22.4.4. 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, <u>and only if</u>, 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.

22.5. Key Signature Dialog

The key signature dialog allows the required key to be entered in one of two ways, which are complementary:

- Specifying the number of sharps or flats in the key and, separately, whether it is a major or minor key. The number is specified in a data entry field, which is usually selected when the dialog is first invoked.
- Specify the required key by name using a drop down control.

22.5.1. 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.

22.5.2. Default Accidentals

Default Accidentals can be associated with a key signature; see section 22.4.2.

22.5.3. 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. E.g, 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.

22.5.4. Octave Shifted Play-Back

In addition to setting a transposing instrument, it is possible to temporarily set octave-shifted play-back, such as 8va and 8vb. Place a hidden key signature before and after the required section and specify the required octave shift (with *Octave Shift* checkbox checked):

- *before* specify +1 for 8va, -1 for 8vb, etc (up to ±3). This adds a temporary octave shift over and above any existing transposition;
- *after* explicitly revert the transposing to what it was before the section (e.g. **0**).

22.5.5. Showing Naturals Before Bar Line

As of Lime 9.17, there is an option to plot naturals at key change before any preceding bar. This is the '*split over bar lines*' checkbox; if it is partially checked, only naturals on mid-system key changes will be separated.

22.5.6. Applying to Selected Staves

When entering or editing key signatures, one must specify whether the change applies to all staves (in the context) or just to the selected ones. This is particularly important if the signature is being inserted to set up a transposing instrument part, or one that is octave-shifted. By default, when invoked, the key signature dialog will default to all staves unless:

- it is invoked with Shift,
- or it is invoked in group select mode,
- or some staves have different key signatures from others;

otherwise it will default to the staves of the selected note(s).

22.5.7. Deleting Key Signatures

As with other symbols, key signatures can be deleted. If one is deleted in a piece, the signature reverts to the previous one. When deleting a key signature, Lime will honour the transposing and graphical settings used in the dialog, from which the delete is performed.

In general, deleting a key signature is not recommended, except in special cases. Normally one should use C major or A minor, which have no accidentals. 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.

If there is no key signature, a part cannot be transposed, except by adding a signature, transposing it, then deleting the signature again! A part with no key signatures is useful for pitch-less percussion, to avoid any possibility of it being transposed.

22.6. Clefs Dialog

In common with the other symbol dialogs (Time Signature, Key, etc), the Clefs dialog allows one to insert or manipulate Clefs before chords or at the end or beginning of the system.

For the '*Before Current Chord*' option, if there are associated grace notes, it normally means before all grace notes or, if deleting, any Clef associated with them. When handling Clefs before chord '*On Current Staff Only*', there is a check-box option to include *Grace Notes*. If this is **not** checked, then insertion/deletion will only apply to the selected note(s) or grace note(s), otherwise it will apply across the chord. It's only shown if there are grace notes and is checked by default.

22.6.1. Clefs 8va and 8vb

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 so modified.

If any other instrument transposition is required, it is specified by the Key Signature.

22.6.2. Percussion Clefs – Drum Mapping

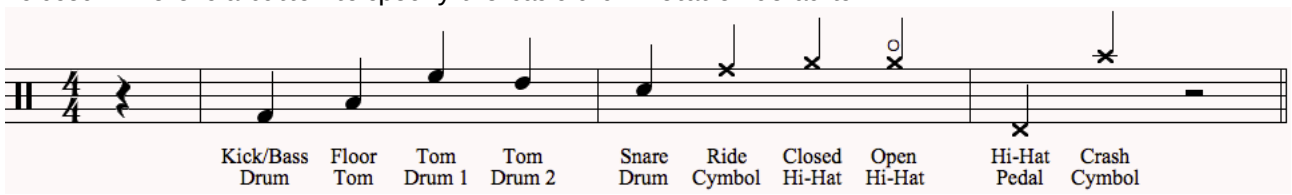
As of Lime 9.18.9, it is possible to specify the actual MIDI pitch that is used for a note on a percussion clef. The percussion clefs are (the most commonly used is the two vertical lines):

In all cases, the middle staff line is nominally middle C (C4).

In the Clefs dialog, if (and only if) the chosen clef is a percussion clef, there is a checkbox and associated push-button to specify what actual MIDI pitch is used for each nominal note (semi-tone). 16 semi-tones either side of the nominal middle C (C4) are supported.

It is only sensible to use this mapping if the voice's MIDI channel is **10** (the percussion channel). Also one should not combine voices on a staff if some have percussion mapping and some do not!

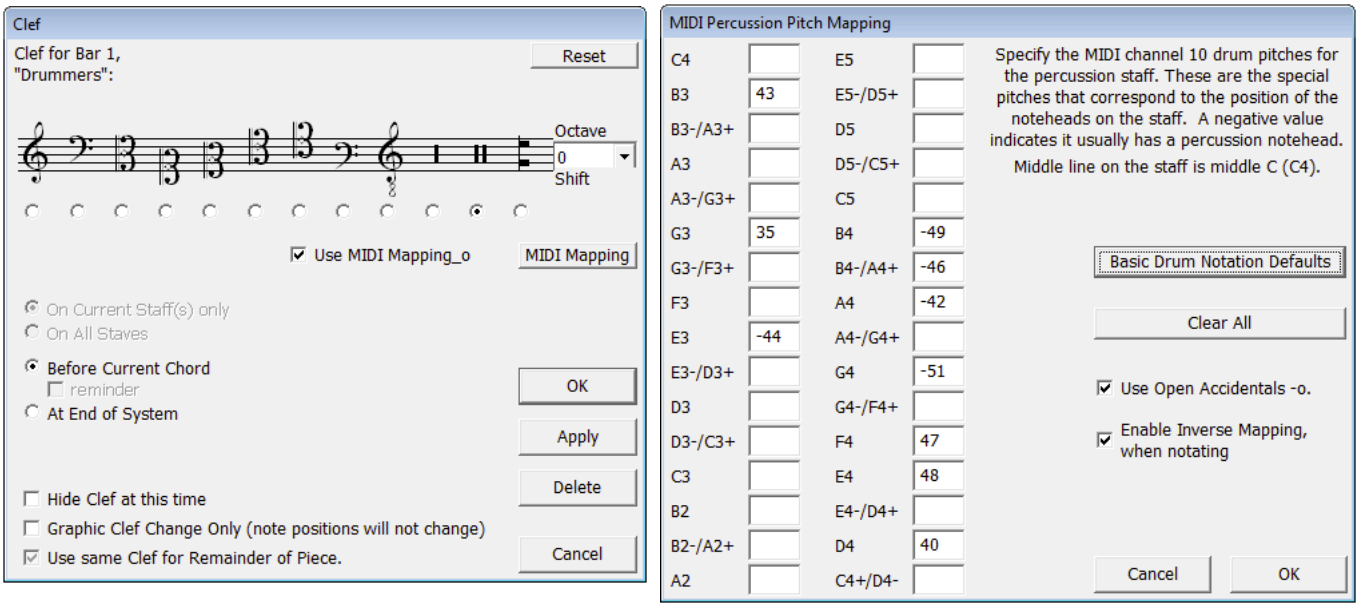
The *Midi Mapping* push-button brings up a sub-dialog to specify the actual MIDI pitch that will be played for each of the specified nominal notes (semi-tones). If an entry is empty (or 0), then the nominal value is used. There is a button to specify the basic drum notation defaults.



Lime can only distinguish sounds if they are a different semi-tone; it cannot do it by note-head; so '*Open Hi-Hat*' is represented by A4 sharp. In the example above, the accidental is a special percussion accidental - the wee O above the cross note-head. This is set using the *Clef Percussion Mapping* dialog.

When a percussion Clef with drum mapping is applied, the pitch of each note, that has mapping specified, will be adjusted. Moreover (as of Lime 9.18.9.1), the notation of any notes, whose pitch is mapped to, will be adjusted as per the mapping.

22.6.2.1. Clefs' Percussion Mapping Dialog



Clef Dialog – with MIDI Pitch Mapping

Apart from setting actual mapping, the MIDI Mapping sub-dialog, has a checkbox '*Use Open Accidentals*' which allows one to specify that a special percussion accidental symbol is used for a Sharp or a Flat. This is set by default if mapping is turned on. The special percussion accidentals[49] are placed above their note-heads (as in the example above for *Open Hi-Hat*) and are always plotted (their show-accidental option is ignored). Furthermore, they affect placement of slurs, ties, and other markings related to the note. There are parameters (under *Accidentals*) that specify the ***open sharp/flat for percussion*** characters. By default, they are a wee O, placed above/below the note-head.

A negative pitch mapping value means that when mapped notes are entered (including MIDI importation) they will default to having a percussion note head (a wee X), as in the example.

22.6.2.2. Percussion Key Signature

The key signature for parts on a percussion staff, particularly if has percussion pitch mapping, should have no accidentals. It is recommended that it should have no key signature; this defaults all pseudo accidentals to Sharp (useful for *Open Hi Hat*) and avoids any possibility of the part being transposed. When setting a percussion pitch mapped clef at the beginning of the piece, if there is a key signature, you will be asked whether the key signature should be deleted (as recommended).

22.6.2.3. Note Entry

The "*Notate as Played on Piano*" preference option (see section 22.4.3) is currently ignored when entering notes on a staff with a percussion clef with MIDI mapping. It will always be *Notated as Played*, and the sound (as played by Lime, if on channel 10) will be according to the mapping[50].

---

[49] The special percussion accidentals are only recognized by Lime 9.18.9 (and later); previous versions will just treat them as normal accidentals.

[50] If you have a MIDI keyboard or equivalent, that can play its own sound, it is recommended that when entering mapped percussion, you turn the keyboard's own sound off, and just let Lime make the sound.

22.7. Transposing

22.7.1. 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 transposes 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 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.

22.7.2. Advanced Note Transposing – Tune Inversion

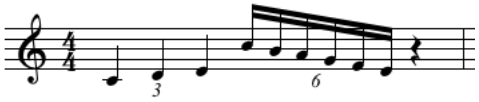
9.17.3 and later offers an advanced facility to invert a tune. This is done by selecting all the notes that form the tune, then selecting the advanced option. This allows one to specify that inversion should occur and the pivot pitch to use. In addition, simple transposition can also be done. If inversion is specified, the algorithm used is:

new_note_pitch = pivot_pitch + (pivot_pitch - old_note_pitch);

The pivot pitch is identified by its **octave** (middle C begins octave 4) and the **quarter tone** within that octave. Note that, even though none of the notes in the tune can be quarter tones, the pivot pitch may need to be and, moreover of course, the result will not create quarter tones.

22.8. 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 shown[51], subject to the *Show Tuplet Numbers* piece option setting.

[51] 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.

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.

22.9. Accidentals

22.9.1. 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. If used with *Shift*, voice based plotting of the accidental will be adjusted appropriately.

22.9.2. Showing Accidentals (by Voice)

In general Lime tries to identify whether a note's accidental should be shown. This depends on the key signature and whether the same note appears previously in the same measure. This can be overridden by selecting a note and choosing the *Note->Show Accidental* menu option. However, as of 9.18.6, automatic plotting of accidentals is recommended.

Lime's logic is voice based, so there is a known issue if there is more than one voice on a staff. Lime will show one voice's accidental, even if another voice has the same note earlier in the measure. Also, if a voice moves to another staff mid-measure, Lime will not take account of the staff change and can omit an accidental. Both issues may be corrected with the *Note->Show Accidental* menu option; however if the staff assignments are different in another context, accidentals may not be correct in that context.

22.10. Tempo Annotations are Metronome Markings

What in Lime is called a ***Tempo Mark*** is often known as a Metronome Marking.

22.11. 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*[52]) while moving the mouse about, a full page cross-hair will be shown.

[52] 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.

22.12. Low Resolution Rendering

22.12.1. 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.

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

22.12.2. 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.

Nowadays, <u>particularly on Macs</u>, copy rectangle is deprecated, because the equivalent can be achieved by taking a screen or or printing to a PDF and using PDF tools to copy selected portions. If a rectangle is copied and pasted into another program, such as Microsoft Word, **<u>all</u>** fonts therein (including **Marl** and **Tufa**) must be installed on the computer displaying the document, otherwise it won't display properly.

22.13. Time Signatures

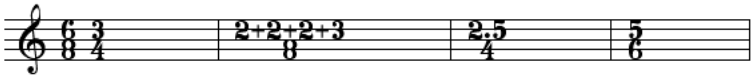
Though, when used with a mouse, they do behave like radio buttons, the Common Time and Cut Time buttons are checkboxes. This means they are not set automatically if selected and, can be explicitly turned off, both of which improve accessibility on Windows. If one of them is turned on, the simple/compound/default radio buttons are all turned off and cannot, therefore, be tabbed to until the Common/Cut time check box is turned off (they can, though, be selected with a mouse).

22.13.1. 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, 6/8 will normally be the equivalent of 3+3 /8 (2 beats in the bar). For common compound signatures, such as 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.

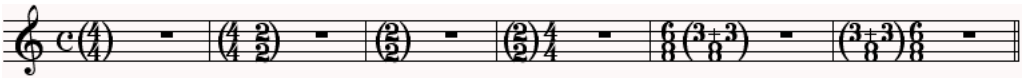


Alternating, Additive, Fractional, Irrational, Time signatures

¿ *Complete section on advanced time signatures*

22.13.2. Time Signature Parentheses

9.17.4 or later supports parenthesizing time signatures, or components thereof (and they are recognized by 9.17.3). In the time signature dialog, if the ***Parentheses*** check box is fully checked, that component will be parenthesized. If the main *Parentheses* is partially checked, then it will have a left parenthesis; if the alternate *Parentheses* is partially checked, then it will have a right parenthesis. This allows the whole of an alternating signature to be parenthesized, or just part of it. Some examples of the possibilities are shown below:



Examples of Time Signature Parentheses

22.13.3. 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 of below the whole system, or be hidden completely.

22.13.4. Beaming Rule

If the *Special Beaming Rule* option is checked, one can specify the beaming rule that is applied after this time signature in all relevant parts. To specify the required rule, click the button that is shown. If it is not checked, the default rule is applied. The rule can, of course, be changed subsequently, using the *Beaming Rule…* dialog (see section 22.44). A partially checked *Special Beaming Rule* option means there was a special rule, which the user hasn't yet changed, but perhaps ought to if the time signature is changed.

22.13.5. 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. Actual entry of the multiplier uses a separate dialog, invoked by the button shown by checking the *Polytemporal* option, that allows it to be entered either as a decimal or, more usefully, as a (heavy) fraction or the product of two fractions - see section 17.5.

22.14. 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.

As of 9.17.4, the *Page->Insert Measures…* dialog also includes an option to insert the measures ***after*** the selected chord. If this is chosen, the new measures will be inserted before the following bar line. This is useful if the bar line is not a single line, but is, for example a repeat or end of section.

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.

22.15. Symbol Placement

22.15.1. Bulk Changes – Systems Option for Symbols at End of System

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.

22.15.2. 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 use will be asked whether the symbol(s) should placed before or after the break.

22.15.3. Clef, Key Signature and Time Signature Dialogs

The *Clef*, *Key Signature* and *Time Signature* Dialogs provide options for controlling placement of these symbols. This can either be before the selected chord or at the beginning or end of a system. Before the chord normally means before the selected chord(s) and all associated grace notes[53].

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.

[53] The one exception is the Clef dialog, which has an option to put the clef in the middle of a grace note sequence.

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*'.

22.16. 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.

22.16.1. Apply

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

22.16.2. Previous, Next and Reset

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 '*Reset'* simply cannels unapplied changes and resets the dialog to the current symbol.

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.

22.17. Saving

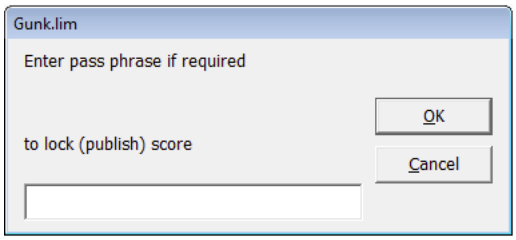
Whenever a piece is saved, it is first saved to a temporary (unique) file in the target directory, which is then renamed to the required file name. This means that, if there is a problem while writing a file, the original file will not be corrupted. Error messages may sometimes refer to this temporary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

22.17.1. 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. Use Shift and choose the *Save as Locked (published)…* item from the File menu. You will then be asked to enter a 'pass phrase', which, if not empty, must[54] be used if the score needs to be unlocked. Shift is required to avoid inadvertently locking a score; on Windows the short cut is Shift+Alt+FH



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.

The primary reason for locking a score is to avoid inadvertently changing it (e.g. when score reading with Lime Lighter). For that, a pass phrase is not recommended and is probably unnecessary.

22.17.2. 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).

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.

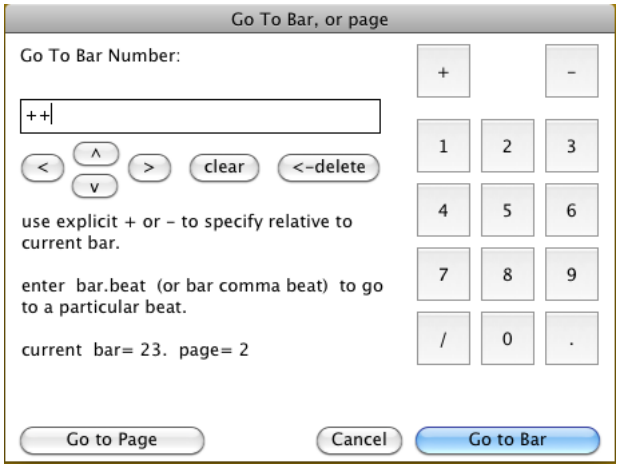
22.18. Go To Bar and Go To Page

22.18.1. Go To Bar/Page Dialog

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 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. The dialogs default to the specified option, but there is an alternative button for going to the other option.

When a specific bar number is used, it is the bar number as printed. If there are bar number disjunctions, a simple *Go to Bar* will only go over a disjunction if the required bar cannot be found. However, *Go to Bar* **00** (two zeros) or *Go to Page* **0** will always go to the start of the piece, whether or not there is a pick-up measure at the beginning and regardless of any bar number disjunctions.

[54] If a pass phrase has been forgotten, a Lime consultant may be able to help.



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.

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.

22.18.1.1. Relative Measure or Page Number

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

Note that the relative measure number is the actual number of measures from the current position, ignoring any disjunctions in bar numbering.

22.18.1.2. Next or Previous Repeat Bar

Go To Bar supports going to the next or previous bar with repeat dots:

+- (plus minus) will go to the next bar with repeat dots;

-+ (minus plus) will go to the previous bar with repeat dots;

these can be followed by a number to go to the Nth next or previous one.

22.18.1.3. Bar Numbering Disjunctions

If the piece has bar numbering disjunctions (bar lines which restart the bar numbering, see section 13.1), it is possible to use Go To Bar to navigate to them:

.+ (period plus) will go to the next disjunction;

.- (period minus) will go to the previous disjunction;

these can be followed by a number to go to the Nth next or previous one.

22.18.1.4. Former Place

.. (period period) will go the bar/page before the last use of the Go To dialog, provided there has been no significant change (such as swapping views).

22.18.2. 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.

22.18.3. 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.

22.19. Selecting a Note

22.19.1. Go Back to last selection

The Note menu option ***Go Back to last selection*** 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. Previous selection is not remembered across view changes.

Short cuts are Windows: *Ctrl+Shift+Space*, Macs: *Command+Ctrl+Shift+Space*.

22.19.2. Note.

The ***Note*** option in the *Note* menu is equivalent to selecting Music Mode in the Piano window (or the *Annotation Creation Mode>Music Mode* option). In addition, it forces selection of just one note (if a group of notes are selected, all but one will be deselected) and will ensure the selected note is visible.

Short cuts are Windows: *Ctrl+Space (+Alt* optional if needed), Macs: *Command+Ctrl+Space*.

22.19.3. Selecting Invisible Notes or Rests

Normally, when a note is selected, the actual note is visible. However, some notes are often not visible, even if not hidden. This can happen, when there is more than one voice on a staff, sharing a rest or note or when rests are combined into a whole- or multi-measure rest. Invisible notes or rests can be selected using the arrow keys.

If an invisible rest is selected, it will be flashed using grey, rather than the actual note head colour; in note entry mode the rectangle will be smaller. This means that, if a note or rest is selected, something will flash.

22.20. Voice and Part Management

22.20.1. 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. It replaces *Staff Drag* as a much better way of printing two staves together

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).

22.20.2. New Part Creation

When a new Part is created, regardless of where in the piece the current selection is, the new staves for the part will initially be shown for the whole piece. If you want the part not to be to be shown at the start of the piece, then after creation, go to the beginning of the piece and specify the part as not-printing. Then go to the system where you want it to start and specify the part as printing.

22.21. 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).

22.21.1. Next Rhythm Error

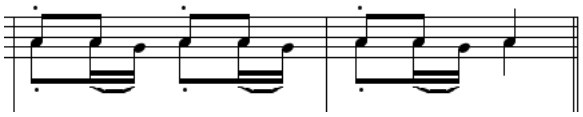
The *Page-> Next Rhythm Error* menu item will go to the next bar whose length is inconsistent with the time signature. This can be useful after importing (or opening) a file with known timing errors.

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.

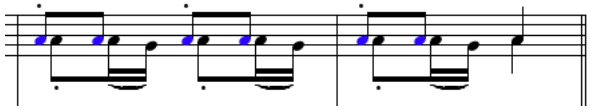
22.22. Don't Share Note Head

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



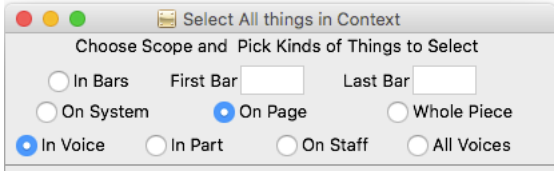
In the example above, the stem up quavers (blue eighth notes) have "*Don't Share Note Head*" set.

22.23. 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.

22.24. Select All Notes or Annotations

There is now a 'Select All' sub-menu under the Edit menu. This allows one to select all notes or annotations in a voice, in a part, on current page or in the whole piece. For selecting all annotations, there is a dialog that allows one to specify the classes of annotation that should be selected (see section 3.24). For selecting all notes there is a dialog for choosing categories of note. Both dialogs require just one note to be selected, from which the scope can be derived.



Choosing Scope and Voices

The bottom row of radio buttons, identify which voices should be chosen; the default is just the selected voice. The remaining buttons identify the scope of the selection, with reference to the currently selected note. If the '*In Bars*' option is chosen, the bar number of the first and last measure must be specified. A blank (or 0) means first measure or last measure in the piece. For both fields a relative bar can be specified by using *+n or *-n, where n is the number of bars relative to the current measure. ***/***+n or ***/***-n can be used to specify relative to the bar of the remembered chord.

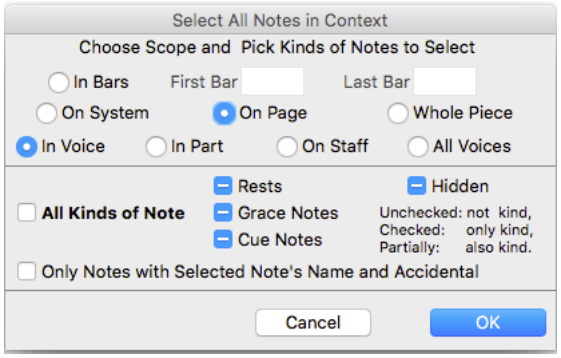
22.24.1. First Bar and Last Bar Special Values

Special values can be entered for the '*First Bar*' and '*Last Bar*' values. As well as allowing specification of the remembered chord, this can make it easier to select the required measures without needing to know the bar numbers.

Code	First Bar meaning	Last Bar meaning
= *(equals)*	First select Remembered chord	Stop after selecting Remembered chord.
. *(period)*	First select Selected note's chord	Stop after selecting Selected note's chord.
<=	Start selecting at beginning of measure containing Remembered chord.	Stop selecting at beginning of measure containing Remembered chord.
>=	Start selecting at beginning of measure, after that containing Remembered chord.	Stop selecting at end of measure, containing Remembered chord.
<.	Start selecting at beginning of measure containing Selected note.	Stop selecting at beginning of measure containing Selected note.
>.	Start selecting at beginning of measure, after that containing Selected note.	Stop selecting at end of measure, containing Selected note.

22.24.2. Select All Notes

For the rare occasions where one does want to select a whole set of notes, there is a dialog, invoked by the *Edit->Select All->Notes in Context* option. By default, all categories of note will be chosen. However, if the ***All Kinds of Note*** check box is not checked, then there are a number of checkbox options to select particular categories of note to select.



Select All Notes

If a kind of note checkbox is partially selected, then both that type of note and its complement will be chosen; if it is fully checked, then only that type of note will be chosen; if it is unchecked then that type will not be chosen. For example to select only hidden rests, fully check both the *Rests* and the *Hidden* checkboxes. The categories are:

- Rests
- Grace Notes
- Cue Notes

The default is that there are all partially checked and, therefore, all notes will be chosen.

The hidden option applies to both rests and normal notes. For normal notes, it means whether or not the note has no note-head.

The '*Only Notes with Selected Note's Name and Accidental*' checkbox allows one just to select notes (not rests) that have the same notated name, octave and accidental as the currently selected note. For example, if the currently selected note was a G4 flat, then only G4 flats would be selected. If it is partially checked, then the octave is ignored; in which case all G flats would be selected.

22.25. Beaming over Rests

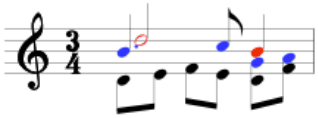
By default, when a beam goes over a **visible** rest, Lime 9.16 (9.16.3) will put a small stem (known as a *stemlet*) on the beam where the rest is. The length of this stemlet is governed by the *stemlet* parameter (under BEAMING). If no stemlet is wanted, then this can be set to zero.

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.

22.26. Multi-Length Staff Chords

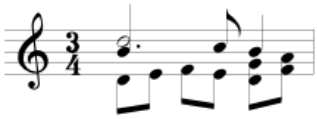
In general it is recommended that, when a staff contains more than one musical voice, each voice is notated as a separate Lime voice. In some cases, where a staff chord contains more than two different note lengths, it may be necessary to have one or more additional voices. The use of multiple voices 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) was created using the '*add*' facility on a single voice, it was 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.

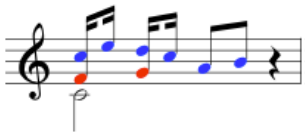
As of version 9.16.5, such notes are no longer discarded; all will be printed. In the following example, all notes are on the same Lime voice (prior to 9.16.5, the half note would not have appeared):



Multi-length staff chord, using 1 voice

Lime refers to Multi-length staff chords in a stem direction as "*Fat Chords*".

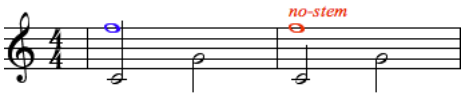
This can, of course, result in some confusion as to what the actual length of a note is. In the above example, it is obvious because the note heads are different. However, if the note heads are the same, it may not be obvious. In these cases, unless *continue primary beaming* is set, by default primary 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. Note that even whole notes have a nominal stem direction, which can affect layout.

If a "fat chord" includes a whole note (or longer) that is not normally stemmed and one or more that is, by default the whole note will be included in the stemming, but slightly offset[55] from the stem. This can be inhibited by explicitly setting No-Stem (*Stem->No Stem*) for the whole notes (if there is more than one in the chord, it will probably be necessary to set it for all of them).



Stemmed and Un-stemmed Whole Notes

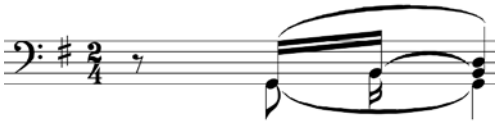
In the example above all notes in the fat, multi-note chords are forced stem up; the red whole note is explicitly *No-Stem*, so the half-note stem is just the normal length. If *No-Stem* is not used, there is a parameter that governs how far the stem extends with respect to the whole note; the default is just a tiny bit, as above; a large negative value (e.g. -99) is tantamount to *No-Stem* on all whole notes. Care should be taken using *No-Stem* if the chord has marks (tenuto, fermata, staccato, etc) or ornaments, because these may be placed at the stem end and could, therefore, overlap the whole note. This is why the default is for the stem to just clear any whole note.

When both stem directions are used in a chord, the layout will depend on which notes are in which direction (remember, though it has no stem, a whole note still has a direction). Normally the whole note(s) will not be forced in the same stem direction as shorter, stemmed notes, so there is no issue about including them in the stem.

Note that a superfluous rest is normally pruned from a chord, unless it is the shortest length object in the chord, when it is important to the rhythm. Like whole notes, if it is included in a chord with stemmed notes all in the same forced stem direction, the stem will, by default, extend to just include the rest. That can be changed by specifying *No-Stem* for the rest.

22.26.1. Accumulated Arpeggios

An example of the use of multi-length staff chords 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 16th B; in both cases make them stem down.

[55] The offsetting of whole notes in fat chords and the *No-Stem* facility is a 9.17 feature. Such chords may not render properly in earlier versions.

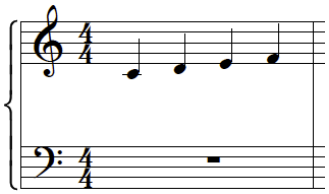
22.27. Arpeggio Ornament Annotations

As well as a simple arpeggio, explicit up and down arpeggio annotations may be specified, using the backslash expressions *arpup* and *arpdown*, or using the ornament annotation assistant. The ornament helper also supports creating big (high) arpeggios, which are essentially two arpeggio characters with a line feed between them.

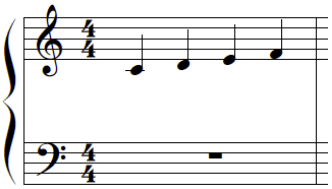
When creating big arpeggios with the helper, there is a checkbox to specify "extended". This currently has no visual effect; however it is used by Lime Aloud and Lime GoodFeel to tell the user or to braille about an extended arpeggio.

22.28. 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.



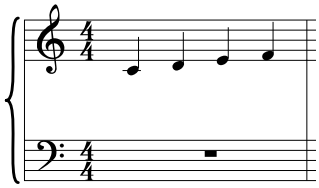
Square Marl/Tufa Braces



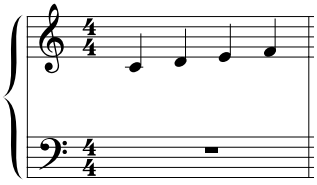
Very curvy Sonata Braces

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.

22.29. Bar Numbers at System Start

In the rare case of a system split in the middle of a measure, the bar number printed at the next system start will be followed by a comma, indicating that it is the number of the bar continued from the previous system.

22.30. 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).

22.31. 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 22.5.3).

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

22.32. Internal Fields

The *Note->Internal Fields* option allows one to inspect and change data associated with an individual note. Changes should only be made if you know what you are doing. Unless Duration Editing is enabled, changes that would affect musical duration are disabled. These include ***lgval***, which is the type of note, the number of dots, and whether the note is a grace note.

The wee checkbox after the '*Plot Acc*' one, is to apply the '*Plot Acc*' setting even if automatically plotting accidentals.

22.32.1. 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.

22.33. 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 follows[56]:

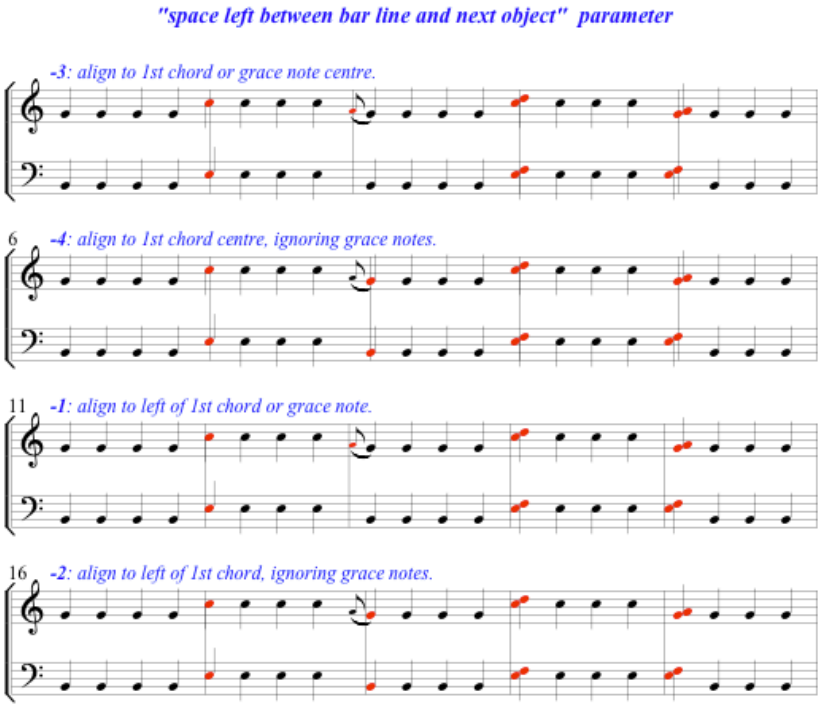
-1: Align bar line to the left edge of the leftmost note head of the first system chord or grace note.

-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):



Aligning Bar Lines with Note Heads

[56] 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.

The bar line through 1st chord facility was introduced in Lime 9.16.4 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 it might be useful to have a parameter 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.

22.34. Align Bars…

The *Symbol->Align Bars…* is a slightly esoteric hack to automate dragging chords such that bar lines before selected notes on different systems on a page are horizontally aligned. There is also an option to make measures on each system in a page equal width. As well as for some modern music, this can be useful for laying out educational material.

It must be recognized that using the *Align Bars…* option is essentially like dragging a set of chords. It can only be cleared by clearing chord drag on selected notes. To clear it:

Use *Edit->Select All->Notes in Context*, and choose the required scope (e.g. *Whole Piece* and *All Voices*),

Then *Edit->Clear->Chord Drag*.

22.35. System Breaks

22.35.1. Re-Pagination (Auto-Paginate)

If the number of systems per page, for any of the options in the **Systems…** dialog, is set to **0** (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.

A simple way of re-paginating a context is to use the **Page->Systems…** dialog and select the option **With Existing Systems, Make New Page Start Locations**, and set **Systems per Page** to zero or blank.

Note that, if the checkbox **No Change** is selected, there will be no change to the pagination, but reminder option for Clefs or Key signatures at the beginning of systems will be enforced.

22.35.2. 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.

22.35.3. 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 17.6), 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 us 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.

22.35.4. Reminders

Whenever the Systems option is applied to a context, the options to show reminder Clefs or Key signatures at the beginning of systems will be enforced, adding any reminders that are needed (or missing) and deleting any that are redundant.

If changes are needed, or if there are any problems, the *Page->Reminders Clefs and Keys* option provides the facility to set or clear reminders quickly, with the option to do it for all contexts; see section 16.3.2 for more details

22.35.5. 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.

22.36. Copying and Pasting Music

22.36.1. 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.

22.36.2. Copying and Pasting Measures

The facilities to copy and paste measures, using the ***Copy Measures*** dialog, has been enhanced:

- The *Copy Measures* dialog includes an option to copy (and paste) key signatures. If this is chosen, the copied material will be pasted exactly as it was in the original. When pasting, if the destination has a different key signature (or octave shift), appropriate key signatures will be inserted before and after the pasted material.
- There is an option to copy lyrics. If this is chosen, all lyric annotations will be copied and pasted (subject only to context issues – see below).
- As of 9.19.3, there is a specific option to copy chord symbols. If this is chosen, all chord symbol annotations will be copied and pasted (subject only to context issues). For backwards compatibility, the option is on by default, if the copy all annotations is on.
- There is an option to copy all annotations, *except titles, parameters and MIDI on the first chord*. If this is chosen any other annotations on the first chord will be copied. If needed, there is still the option to copy all annotations except those on the first chord. Note that copying lyrics or chord symbols is independent of both the first note options; they are either copied or not.
- For copying annotations (or lyrics or chord symbols), there is an option to specify whether those in *Any Context* are copied. If not chosen, annotations (and lyrics) that are specified for a single context other than the Score or the current context, are not copied; though all annotations specified as *Only–in–Score* and *Not–in–Score* are always copied. If chosen, all annotation in any context are copied.
- If the selected chord is not at the beginning of a measure, there is an option to actually start at the beginning of its measure. This is always set by default.

If the ***Copy Measures*** dialog is invoked with the shift key down, it will default to all following measures and, if the current default is just the selected voice, will default to all voices on selected staff.

Copying measures of more than one voice (all voices on selected staff or all staves) is not possible if there are voices that are not aligned at the selected position.

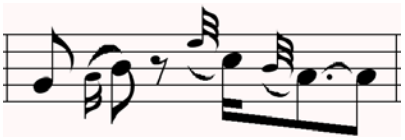
22.36.2.1. Copying Measures Between Contexts

Currently, if you Copy Measures using All Voices on Staff (or All Voices) in one context and paste them into another context, you will probably need to reset what voices are printing on the staves of the destination context. This also applies if the destination is actually a different piece.

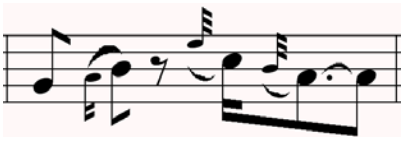
22.37. Note Stem Flags

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.

22.38. New Note Heads and Accents

Lime 9.16.9 added a built-in Rhythm/Slash note-head ( ) and an anti-accent ( ).

22.39. 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*".

22.40. Grace Note Rests

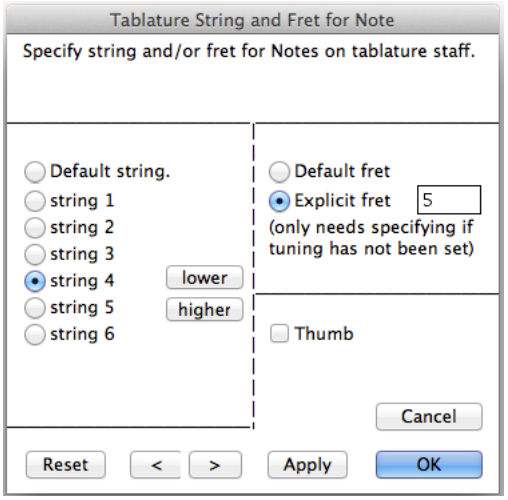
It has always been possible for a grace note to be a rest and, moreover, hidden. Normally, when hearing, it will have the same length as all other grace notes in the sequence. However, as of 9.17, if the grace rest is also specified as a 'Cue' note, it will default to having no length at all and, moreover, can nominally be any tuplet (which, if hidden, can be useful for anchoring a tuplet bracket).

22.41. Guitar Tablature

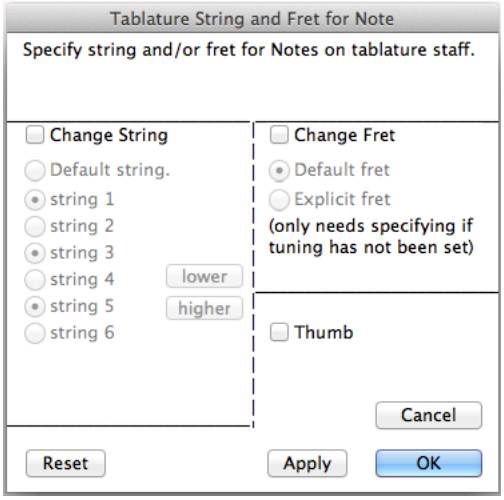
By default, Lime will automatically assign string and fret based on the prevailing tuning. The default tuning is the standard guitar 6-string tuning. Currently the tuning can be changed using the "***MIDI pitch for string n in guitar tablature***" parameter set under "***Tablature Pitches and Symbols***" (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.

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.

The default is based on the specified tuning, which is set using the **Tablature Pitch** parameters. For anything other than 6-string classical guitar, the tuning will probably need changing.

22.42. Default Accidentals… Menu Option

If more the one note is selected (group selected) the *Default Accidentals…* menu option will apply the defaults to the selected notes and any chord symbols on them. If in text annotation mode and more than one text annotation is selected (group selected) the *Default Accidentals…* menu option will apply the defaults to any chord symbols that are selected. Of course, an annotation must have category *Chord Symbol* to be treated as one.

22.43. Swing Eighth Note Pair

The swing eighth note pair option has been enhanced in two ways:

a. Valid settings are remembered if the playback is re-computed. Explicitly clearing the option is needed in order to remove the swing. Prior to Lime 9.17.2, re-computing playback would simply remove the swing.

   If shift is used when selecting the menu option, the user can set whether to set or clear the swing and can specify the scope to which the change should apply – selected notes, selected voices or the whole piece.

b. The option can be turned on (and off) for note entry. If selected when entering notes, valid swing eighth combinations will be swung automatically. The option is on by default, if the first selected chord to be replaced was part of a swing eighth pair.

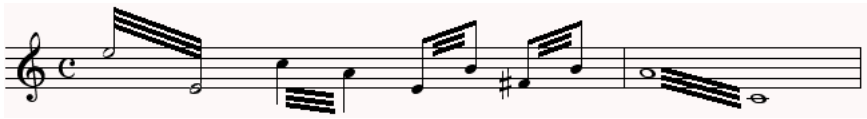
22.44. Beaming Rule

The prevailing beaming rule specifies the default beaming boundaries in 8th notes (*quavers*) from the start of a measure. When a chord in a beam ends at a beaming boundary, the beam will be ended. In general, the beaming rule is associated with the time signature. As of 9.17.8 it can be specified whenever a time signature is inserted or changed (before then the default was always applied). The rule may be changed anywhere by selecting a chord and using *Stem->Beaming Rule…* It is possible for different voices to have different beaming rules.

As of 9.17.8, the number of eighth notes specified in the *Beaming Rule…* dialog may be fractional. So, for example, 1.5 (or 3/2) eighth notes means 3 sixteenth notes (*semi-quavers*). Earlier versions of Lime fully support the resultant rule (they just do not have the ability to enter it)!

22.45. Tremolo Beaming

Tremolo Beaming notates rapid alternation between two chords[57]; some examples are shown below:



Examples of Tremolo Beaming[58]

[57] As of version 9.19.6, when hearing, Lime will rapidly alternate a tremolo beamed chord pair.

[58] The old manual does describe how to tremolo beam (around page 95), but not very clearly.

To create tremolo-beamed pairs as in the examples above, one needs to enter a tuplet pair of chords, using the type that, if beamed together would have the required number of beams (in all examples a 1/32nd note was used). The tuplet should be 2 in the power of 2 necessary for them to occupy the required space. The power of 2 specifies how much longer the printed note actually is (1.5 times a power of 2 can be used if dotted notes are wanted). Having enter the chord pair, select all notes in the pair of chords and use:

Stem->Tremolo Beam.

To get them beamed, it may then be necessary to select the first of the pair and use:

Stem->Continue Primary Beam.

The separation of the tremolo beaming from the note stems is governed by parameters (under *Beaming*). By default, if the note head is a *1/2* note, there is no separation and the beams extend up to the stems (but this can, of course, be changed). When the printed notes are whole notes, the appearance can be slightly dependent on whether they are nominally stem-up or stem-down.

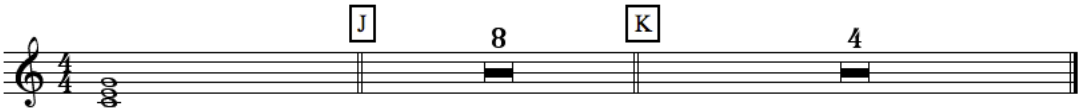
22.46. Enabling Combined Rests

All things being equal (e.g. all but the first have no annotations), consecutive rests will be combined to make whole measure or multi-measure rests. This can be prevented by explicitly specifying that a rest should not be combined. This is done by selecting one or more rests and using the ***Note->Don't Combine Rest*** menu option; check it to prevent and uncheck it to allow combining.

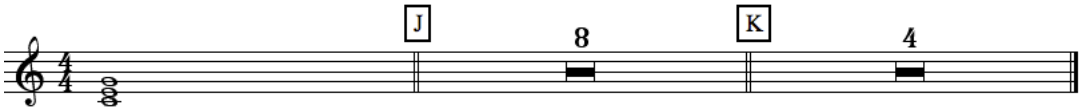
22.47. Multi-Measure Rest Bar Width

Lime 9.17.6 and later supports a context option to specify the algorithm used to determine the width of a multi-measure bar[59]. In the *File->Options for Context…* dialog, this is the *Bar Sized Multi –Measure Rests* checkbox. There are 3 options:

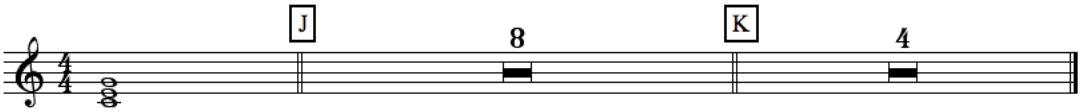
- Checked — All multi-measure rest bars are sized based on the length of a **single measure** in the specified time signature.
- Unchecked — All multi-measure rest bars are sized, based on their actual **musical length**, so that those with more measures will be wider.
- Partially Checked — Multi-measure rest bars at the beginning or in the middle of a system are sized based on the length of a single measure in the specified time signature, but those at the end of a system are sized based on their actual musical length.



Partially checked – Multi-Measure Rests are Single Bar Sized, Except at End of System



Fully Checked – All Multi-Measure Rests are Single Bar Sized



Unchecked – All Multi-Measure Rests are Sized based on Musical Length

The partially checked option is for backwards compatibility because that is what Lime, prior to 9.17.6, always did. After 9.17.7, the default is for musical length sizing.

Of course the actual widths are subject to the surrounding music, symbols, annotations, etc, particularly at the end of a system. Musical length sizing of multi-measure rests uses the same logarithmic algorithms as are used for spacing ordinary notes/rests. This means that, normally, the greater the number of measures, the greater the width (albeit, often only a little bit greater).

As of 9.17.7, if the *Linear Spacing* option is set, all proportional sizing of multi-measure rests is effectively disabled; they are just a little bit larger than an ordinary measure. This is because otherwise they could easily squeeze all other measures in a system far too much.

22.48. Inserting Symbols before Combined Rests

If one group selects a rest that is actually a combination of more than one rest (whole measure or multi-measure) and then inserts a symbol, such as a Clef, key or time signature or even a bar line, Lime (9.17.9 and later) will only insert at the beginning of the group if the whole group is selected.

[59] The Multi-measure rest option is understood by Lime 9.17.5 but not by any earlier version.

22.49. 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 Read and save the selected Lime score files. The files are automatically updated to the current file format. Options are available to:

- 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 *Score* context.
- On Windows, the default character set to use if the character set is not specified for an annotation. The default is the computer's code page.

The primary function of Batch Update is to update an old set of Lime files to the latest format. This is particularly recommended for Lime 9.17.

Export MusicXML Read and export the selected Lime score files to MusicXML. N.b. Lime's exporting of MusicXML is fairly primitive (see section 7.1.12).

Export Midi 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.

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.

22.49.1. 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 tilde.

22.50. Lime Console

For certain operations[60], Lime will write log and status information to a 'console' window, which will usually (but not always) be shown automatically.

On both Macintosh and Windows, there is a '***Show Lime Console*** item in the Window menu. If the console is not shown, invoking the menu item will show it, otherwise the menu item will hide it. Invoking it with the **Shift** key pressed will always show the console, position and size it to the default, and bring it to the front,

On Windows it is a standard Windows' console with its own menu (Lime logo on its top left). One can change its characteristics, such as font size (which will be remembered).

[60] For example, the Batch… facility uses the console

22.51. Remember Voice Chord

As of Lime 9.19.2, there is a facility to remember the currently selected note (chord) and then select another note in the same voice and do various things between the new selection and the remembered chord, such as selecting all notes or drawing a line or curve.

The currently selected note can be remembered using the ***Note->Remember this Voice Chord*** menu item (*cmd/ctrl-***=**). The selected note will be remembered, while the current view is active (and, of course, as long as the note still exists). It will be forgotten if the view is changed. When there is a remembered note, the menu item will be checked; it will be fully checked if the current insertion point is the remembered note, otherwise it will be partially checked.

If you switch to another context in the same piece, the chord remains remembered, providing its voice is active in the context.

22.51.1. Selecting Remembered Note

When there is a remembered chord, 3 selection options will be activated:

- *Note->Go to Remembered Chord*
  Select the remembered note.
- *Edit->Select All->Voice Chords from Remembered, excluding last*
  Select all notes between the remembered chord and the currently selected note, excluding the last chord. This can be useful, for example, for setting slurring.
- *Edit->Select All->Voice Chords from Remembered, including last*
  Select all notes between the remembered chord and the currently selected note, including the last chord. This can be useful, for example, for transposing or setting note head characteristics.

For these options, the remembered note and the currently selected note **must** be in the same voice.

There is also a facility to duplicate annotations that are on multiple notes, to the remembered note and to the notes following it (see section 3.26).

22.51.2. Curve or Line Drawing

There are Line and Curve build assistants to draw special Lines or Curves between the remembered chord and the currently selected note, providing the currently selected note is in the same voice as the remembered note, or there is a note in the selected note's staff chord that is in the remembered note's voice. If the current context is the score, the annotation will be in all contexts; otherwise it will just be in the current context. The resultant annotation is a normal annotation and handled and may be modified like any other one.

Currently (9.19.2), there is only an assistant to draw a Phrase Indicator curve (see section 3.17.2.2). However it is planned to implement others (e.g. normal curves, hairpins, or brackets).

Menu item *Annotation->Line and Curve Build Assistants->Draw Phrase Curve from Remembered Chord* will draw an extended, simple curve (not fattened) between the remembered chord and the voice chord of the currently selected note. The beginning will be at the first chord note and the end will be at the last chord note. When drawn the end will be selected.

If the Shift key is used when selecting the assistant, the curve will be downwards, below the staff, otherwise it will be upwards, above the staff. The curve will be horizontal and the assistant attempts to draw the curve above or below the highest or lowest chord ornament.

22.52. Shift Key Menu Alternatives

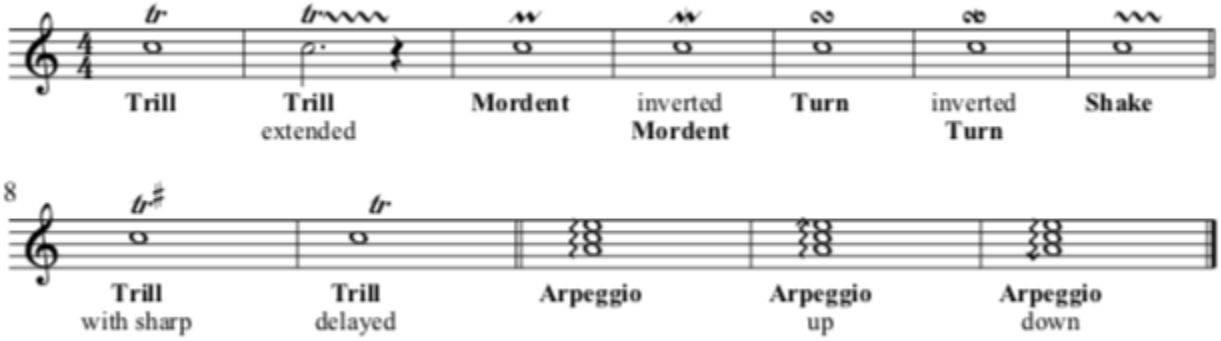
The Shift key is often used with menus (or equivalent) to modify slightly what the menu actually does. The following is a list of menu items affected by the Shift key.

- *Next* and *Previous Page* become First and Last Page.
- When Shift is used, *File->Revert* is available, even if the file is apparently unchanged; also the file will revert to the saved context and page. If Shift is not used when invoking *File->Revert*, the file will be re-opened in the current context and page.
- 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 Window, it defaults to very **high** resolution (changeable in the print dialog).
- 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 *Key Signature or Time Signature*, the default will be forced to the selected staves; otherwise the default will be all staves unless in group select mode.
- 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.
- When Shift is used when deleting a single text annotation, a new text annotation will be started (like it always was, prior to Lime 9.17.2)
- When Shift is used with *Text->Model for Category*, the status of the selected text annotation will be flipped, otherwise one is always asked what to do.
- When Shift is used with *Note->Show Accidental*, it will also flip the *even if automatic* setting.
- When Shift is used with *Window->Show Console*, the console will always be shown.
- When Shift is used with one of the *Annotation->Annotation Creation Mode* = items, it behaves exactly like clicking the equivalent piano window icon; otherwise any selected annotation will be deselected.

¿ *Complete list of Shift modifications to menus.*

22.53. Note Ornament to Play

Lime 9.20 will support the ability to specify that a Note has an ornament that will be played automatically when the piece is heard. These ornaments are notated without using annotations. The supported ornaments include **Trill**s, **Mordent**s, **Turn**s and Arpeggios.



Note Ornaments to play Examples

These *Ornaments to Play* use a slight change in the file format. This change is recognized by Lime 9.16.6 (and later), but not by any earlier version of Lime. <u>However, Lime 9.16.6 can only manipulate such ornaments in files that were created by Lime 9.20 or later</u>.

Creation and manipulation of *Ornaments to Play* s done using a new menu item:
Note->Ornament to Play....

A note can only have one *Ornament to Play*. If required, an *Ornament to Play* can be specified as not to be notated. This can be useful if a special annotation is required.

An **Arpeggio** is special in that it applies to the whole voice chord. When it is set, Lime will ensure the chord is in the correct order (sorting it if necessary), and will set it on the 1st note in the chord.

22.54. Lime Score Format with Graphics

¿ ?

23. 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.

23.1. OSX Version Support

Lime 9.16 (and later) 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 9.19.

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.

23.1.1. Catalina, 10.15, **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.

Google "***32 bit apps on Catalina***" for more information and possible work-arounds, e.g. https://uk.pcmag.com/old-news/123012/how-to-run-32-bit-apps-in-macos-catalina.

It is, however, still possible to run Lime on MacOS 10.15 or later by using a virtual machine (e.g. ***VMware Fusion*** or ***Parallels***), running Windows or an older MacOS. Google "*32 bit Apps Mac*" for more information. It is worth noting that WINE wrappers, such as **Porting Kit** (free) or **Crossover** can be used to run the basic Windows' version of Lime on any Macintosh that runs on Catalina or later. See section 23.10 for more details.

It's also worth noting that the release version of Lime is still compiled and verified on a Snow Leopard (OSX 10.6) virtual machine, on which it runs very nicely.

23.1.2. 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 23.2.2), 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).

23.1.3. Yosemite, 10.10

There's a problem 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 or High Sierra!

23.2. Printing

Since version 9.15, high quality printing from Lime is available on all supported Mac platforms.

23.2.1. 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.

23.2.2. 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 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.

23.2.3. 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. Also, 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.

23.3. 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***[61] from http://notahat.com/simplesynth is very strongly recommended.

23.4. Fonts Dialog

If a font is required that is not one of Lime's standard ones, the ***Annotations->Font->Fonts…*** 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. This simplified fonts dialog can be invoked on any Mac by using the Shift key when invoking the ***Fonts…*** menu. If also invoked with the ctrl key (mac version) the 'simplified' dialog will use exotic script for some of the font names.

23.5. 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.

[61] In spite of the warning on the web site, ***SimpleSynth*** works perfectly well on High Sierra.

23.6. 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 possible[62]. 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.

23.7. Maximizing

The *Window->Full Size* option, makes the current music window as high as possible on its screen and positions it to the left of the screen. This is also available by clicking on the window's green maximize button (top left of window).

If the music window's maximize button is clicked with the *Alt* key pressed, the window will be enlarged and zoomed to fit the screen width. Clicking it again (without Alt key) will revert to its previous zoom.

23.8. Music or Symbol 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 PopChar[63], 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.

There are known issues on Macs editing an annotation that contains more than one font, one of which is a music or symbol font. Main issue is specifying a different font to enter new text in the middle. For this reason, such annotations will use backslash expressions for editing, unless explicitly disabled.

Because of its antiquity, it is recommended that the Symbol font is not used, unless absolutely necessary. There should always be appropriate unicode symbols in a normal font.

23.9. 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). Note that if it is done while Lime is running, one will either have to use the Hear->Midi Reset… option (or resetart Lime) for Lime to see the new device.

---

[62] Actual limitation is 4095 pixels in each direction

[63] www.ergonis.com/products/popchar

23.10. Use of WINE

Windows Lime will run under **Porting Kit** (free) or **CrossOver** on any Mac. They work **on Catalina** and later (and earlier), including the modern silicon Macs. It is worth noting that these are wrappers round WINE, which is a compatibility layer capable of running Windows' applications on Macs. These were primarily developed for running games. There are other wrappers (such as PlayOnMac), but many are no longer supported.

23.10.1. Installation

You should use the special Windows' ***Lime.wine.exe*** installation package to set Lime up or update it.

After that, one can also update Lime by simply copying new *Lime.exe* or other files into the drive_c folder used by the wrapper.

23.10.1.1. C-Drive on Porting Kit or CrossOver

On Porting Kit or CrossOver, if you right click the bottle, there is an option to open the C-drive.

The wrapper's 'virtual C drive' will be in inside the created Lime application. You need to select the application *~/Applications/appname* then choose *Show Package Contents* then you should see an alias to the **drive_c**, then look in the Program Files for Lime.

23.10.1.2. C-Drive on other Wrappers

PlayOnMac's 'virtual C drive' will be:
~/Library/PlayOnMac/wineprefix/Lime/drive_c

If your Library directory is not visible on your Mac, open the Terminal app and type:
chflags nohidden ~/Library/

23.10.2. Issues

You need patience when starting Lime (or another program) as it takes a very long time, with no feedback, to load things. Once loaded, Lime is reasonably speedy.

23.10.2.1. Printing

Wine on new Macs (certainly ARM ones) on Sonoma or later do not seem to support printing to a printer! This is probably because Macs no longer support postscript (they say for security reasons) and aolder versions of Wine do not cater for that. This is fixed in CrossOver 25, which runs Wine 11; so it'll probably be fixed in other wrappers sometime; but hasn't yet been in May 2026.

The special Lime for Wine includes the capability to print to a PDF file. If normal printing doesn't work, this can be used to print to a PDF file. This is done by lime invoking a wee app to convert the PostScript file created by Window into PDF. For that, the emulated Windows' version **must** be **10** or less. If the emulated version is 11 Windows 11, Wine creates an invalid PostScript file, that cannot be converted.

23.10.2.2. Font Fall-Back

There does not seem to be any font fall-back (not even Apple font fall-back). So if unicode characters are used with a font that doesn't have substitution specified, they are not displayed.

This seems to be resolved (spring 2026) with the latest CrossOver and Wine 11. Let us hope that Porting Kit and other wrappers will update soon.

23.10.2.3. Closing

Closing Lime/Wine kills the application, without any save warnings! This is a major issue, as it is easy to close Lime/Wine and lose changes. You should use the menu item.

23.10.2.4. Control and Alt

The latest Posting Kit and CrossOver, seem to map the Command key to CTRL; so you can use either.

On PlayOnMac, ***Ctrl*** has to be used for Control; ***Cmd*** has to be used instead of ***Alt*** (including to invoke menu items).

23.10.2.5. Windows Version 11 Problems

As documented above, there are currently (spring 2026) problems with Wine using Windows version 11 or later. So ensure your WINE's Windows' version is set to Windows 10 or earlier (XP or later).

23.10.2.6. Minor Issues

- Control characters (e.g. Tab) are not recognized as white space. They are treated as though there was no glyph and displayed as a square box. Lime uses Tabs, etc as delimiters in window titles and dialogs, because they cannot be in a file or part name! However that's only a minor issue.
- On PortingKit, one cannot easily set the Mac up to invoke PortingKit's Lime to open a Lime file. One has to explicitly open it from within Lime.
- CrossOver (and PlayOnMac) has a facility to specify that .lim files are opened by it.

It's worth noting that Lime has also been tested briefly on other WINE wrappers. They had the same issues as identified above (so the issues are probably WINE oriented).

24. Windows Considerations

24.1. Windows Version Support

Lime has been tested on all systems from Windows XP up to and including Windows 10. It is being used on Windows 11, and as far as we know, it works well.

Note, however, that support for Windows 2000 is deprecated (though it still works) and is unlikely to continue after 9.19; please contact us if this is a problem.

24.2. 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.

24.3. Do Not: Use Unicode UTF-8 for worldwide language support

The 'Beta Unicode UTF-8' system locale, released by Microsoft in 2022 stops Lime working. Do not use this locale.

24.4. High Contrast Theme

If the Windows desktop is configured to use a high contrast theme, Lime's music window will use the theme if, and only if, the *music window appearance* setting in the *general preferences* is set to ***Windows setting***.

24.5. Multi-Monitor Environment

24.5.1. 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.

24.5.2. 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 menu[64]. Select *Maximize* and Lime should appear on the primary monitor. If *Maximize* (or *Restore*) is selected with the Shift key pressed, Lime's frame window will be moved to the top left of the primary monitor, but not maximized.

[64] ***Alt Space*** is a standard Windows short cut for the system menu.

24.6. 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:

direction	***short cut***	***normal function***	***hear tracking***
Left	*cmd/ctrl*-[	Go to previous page	Slower
Right	*cmd/ctrl*-]	Go to next page	Faster
Up	*cmd/ctrl-shift*-[	Go to beginning of piece	Pause/Resume
Down	*cmd/ctrl-shift*-]	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.

24.7. 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 screen[65] (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.

[65] On Window 7 this option is in: Control Panel->System->Advanced System Settings->Performance->Visual Effects

24.8. PDF Printing

There are issues on Windows with printing to PDF with many of the free PDF creators. Many have problems with large (multi-part, many pages) Lime score at very high resolution. For this reason, very high resolution is not the default, but can be set in the print dialog.

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%[66], for which it works nicely.

At very high 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 not using very high resolution, 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***. So, if, for any reason, very high resolution is needed, It is recommended, that if ***Adobe PDF*** (which works very nicely, but is expensive) is not available, ***doPDF*** should be used for full resolution PDF creation. There is a basic free version and it can be downloaded from:

https://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 (e.g. *doPDF*).

So, when printing to PDF on Windows, do not use very high resolution or use *Adobe PDF* or *doPDF*.

The option to print at very high resolution is only retained on Windows in case someone identifies a need for it. Normally there is no discernable difference, except that PDF files are slightly smaller.

If printing to PDF it is recommended that Lime's fonts are installed in the system, because some PDF printers, notably ***Adobe PDF***, default to relying on system fonts and ignore embedding.

24.9. Ctlr+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 AltGR[67] by itself and with Ctrl.

24.10. 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.

---

[66] Awork-around to this absurdity has not yet been identified.

[67] Windows actually automatically adds Ctrl to AltGr!

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.

24.11. File Name Problems on Windows

24.11.1. 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.

If you are still using these old versions, upgrade to Lime 9.19!

24.11.2. Leading Spaces in File Names

Windows seems to prevent one saving a piece with a file name starting with one or more spaces – the spaces are removed. However, that is hardly a problem!

24.12. Dialog Appearance

With the exception of file related dialogs, all Lime's dialogs are configured to use the font called **MS Shell Dlg 2**. This is mapped by Windows to a real font (the default is usually Tahoma). Should you want a different or bolder font, the font mapping can be changed using **regedit**.

- Go to HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\FontSubstitutes
- Double-click MS Shell Dlg 2 and change the font.
- Close regedit.
- Log out and log in and check that the font is ok.

24.13. 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*.

24.14. Lime for Other Languages

Lime for Windows does support user interface (menus, dialogs, etc) in languages other than English.

The ***PhrasesBlank.txt*** file, which is included in the release material, contains the current list of phrases used by Lime in its user interface. Support for another language is achieved by translating the phrases in PhrasesBlank.txt, putting each translation in double quotes on the line below the original {before the closing curly bracket}, e.g:

```
{ "Lazy",          // original English,
  "Pigrone" }      // translation.
```

A new phrases file is then created with another name, e.g. *Phrases_IT.txt*. If this new file is then put into the same folder as Lime.exe, Lime will use it to provide the user interface in its language.

Please contact us if you are interested in assisting in translating. There is a tool, ***LimeXlat***, which assists in managing translation files, which we can provide.

25. Lime Lighter

Lime Lighter is a Windows[68] 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.

25.1. Manual Scrolling

Low vision tracking (scrolling) is controlled using F13 (go back), F15 or (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 25.7. 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.

25.1.1. Arrow Keys

Because some devices, notably the Airturn pedals, generate arrow keys, rather than function codes and cannot be changed easily, Lime can be configured to map arrow keys to function code:

Up Arrow **F13**, *Left Arrow* **F14**, *Down Arrow* **F15**, *Right Arrow* **F16**.

Which arrow key a particular pedal generates is, of course, device dependent. Lime's mapping is based on the old Airturn BT106 configuration.

[68] Lime Lighter does, by and large, also work on old desktop Macs (but without gestures), but not on iPads.

25.1.2. 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.

25.1.3. 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.

25.2. 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 25.8). 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 4.2.2), 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.

25.3. 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.

25.4. 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.

25.4.1. 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**[69]. 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.

25.4.2. 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.

[69] It still hasn't been fixed in 9.19!

25.5. Mark-Up

Within Lime Lighter on Windows, 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.

As of Lime for Windows 9.18.8.2, when mark-up is invoked from Lime Lighter, the music window zoom will be the same as for Lime Lighter[70]. This means that the music will be approximately in the same position.

25.5.1. Mark-Up Zooming

On Windows when marking-up is active, the mark-up menu includes the ability to zoom-in or zoom-out (make the zoom bigger or smaller). The change in zoom is temporary and only for the mark-up; it will revert when marking-up is stopped.

25.5.2. Mark-up Graffiti

Prior to Lime 9.17.4.2, the colour of mark-ups was always subject to the theme configured for the Windows' desktop. If this were a high contrast theme, the mark-up colours would be white if the desktop were black (and vice versa). This meant that they only worked properly if Lime's music window background was similar, because mark-ups would be drawn in white, regardless of their colour. If the music window appearance was not set to ***Windows setting*** in the general preferences (or was not *white on black color inversion*) and a black background theme was selected, the white mark-ups would be nearly invisible.

Lime 9.17.4.2 and later has a mark-up preference option to control, whether mark-ups are drawn using high contrast. If it is turned off, mark-ups are drawn in their specified colour, whether or not a high contrast Windows' theme is set. This means that a user can have the *Lime default* or *Black on Yellowy* music window and mark-ups will be always be shown properly.

So if mark-up graffiti are hard to see, try turning ***off*** the ***Support High Contrast*** mark-up preference[71].

25.5.3. Printing Marked Up Pieces

By default, when a context is printed, mark ups are includes. 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.

---

[70] In earlier versions, zoom reverted to the basic zoom setting, which was usually smaller!

[71] For backwards compatibility, the ***Support High Contrast*** preference is on by default, so needs to be unchecked.

25.6. Lime Lighter Specific Short Cuts

The ***Esc***ape key always exits Lime Lighter mode and mark-up Draw mode. The following short cuts are specific to Lime Lighter:

cmd/ctrl-` On Windows (ctrl+reverse quote), with Lime Lighter, sets mark-up Draw mode or, with the *shift* key, turns off draw mode

cmd/ctrl-**** On Windows (ctrl + backslash), with Lime Lighter, turns on, or with the *shift* key off, 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 *AirTurn*™ 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.

25.7. 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 — Tap and Hold on a touch screen:
Always exits Lime Lighter music stand and returns to normal editing mode. It is the equivalent of the *escape* key and is **not** configurable.

Shift-Left-Click[72] — Tap with two fingers simultaneously:
By default will bring up a big, touch screen friendly 'Go To' dialog (for bars), but may be configured to stop/mark–up (if needed for short term compatibility) or to do nothing. Note that two finger tap brings up the same enlarged Go To dialog (see section 22.18.1) in normal editing mode, so it is recommended to keep it the same for Lime Lighter.

Left Click — Tap with one finger:
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/mark–up 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 — Tap twice quickly with single finger:
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.

[72] N.b. shift-left-click is only equivalent to 2-finger tap when the music stand is active.

25.7.1. 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 is essential for one finger taps if you use navigation gestures. It may also 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 (see section 22.18.1). 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/Mark-up Stops scrolling and turns Mark-Up on. A pedal press or equivalent is needed to resume scrolling.

It is recognized that there is a need to add a "Go To Rehearsal Mark" capability.

25.7.2. Press and Tap Gestures

In Lime Lighter mode, the Press and Tap gestures, known as "roll-over"(press one finger onto the screen and then quickly tap with another) can be used instead of the pedals (if you have a hand free) as follows:

Direction	*Lime Lighter Scrolling mode*	Action
Right	*manual:*	Advance;
	automatic:	Faster.
Left	*manual:*	Go back one;
	automatic:	Slower.
Up	*manual:*	Take Repeat (or disjunction);
	automatic:	Pause/resume.
Down	*manual:*	Take secondary disjunction (the 'Go To' function');
	automatic:	Pause/resume.

25.8. Lime Lighter Preferences

There is a special set of preferences for Lime Lighter. These enable configuration of:

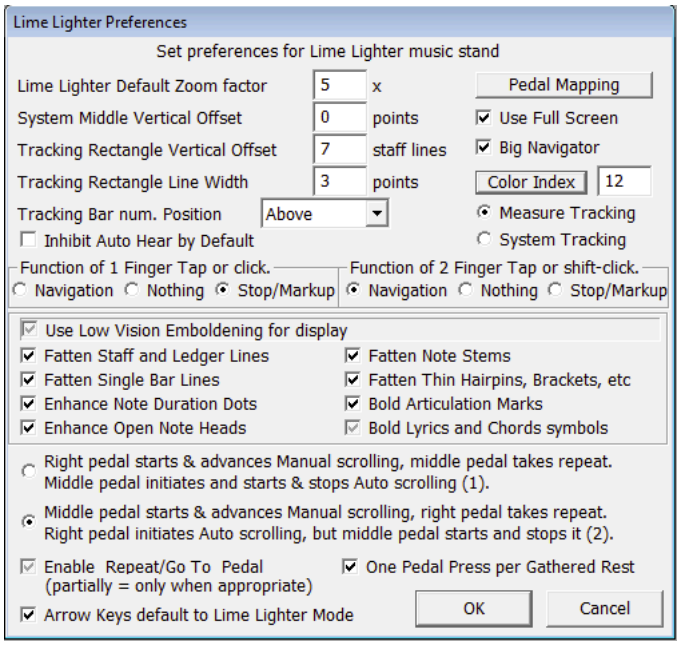
- Zoom to use when Lime Lighting a piece. If this is specified (non-zero), whenever low vision scrolling is invoked, the zoom will change to this value and revert afterwards. **+** or **-** may be used for the zoom factor to specify screen/frame width.

  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. Note that zoom level 10 is ten times the small size, which is about eight times normal size.
- Appearance (size, line-width and color) of the tracking rectangle used when scrolling. Note that this is the same as the tracking line used in normal hear tracking. The tracking bar number position specifies where the bar (measure) number is shown relative to the tracking rectangle.
- 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

25.8.1. 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
Right The function of the 'middle' and '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.

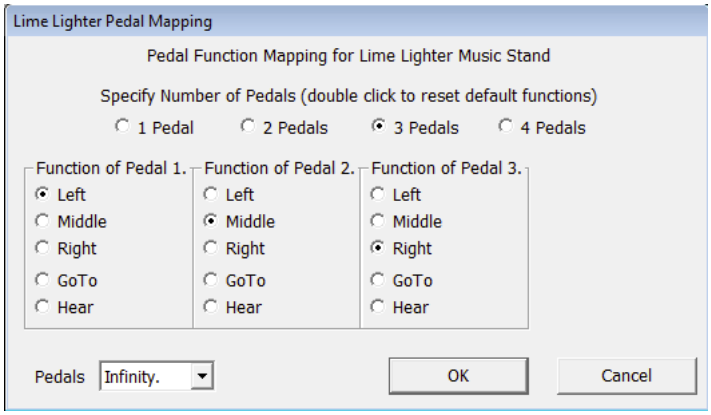
On top of this, you can map the physical pedals on the AirTurn pedal block to the functionality of 'left', 'middle' and 'right'. These are the port numbers on the "digit" box and are numbered 1 to 4 (if you have a 4 pedal device); n.b. a sustain pedal would be plumbed into one of the ports. So if you 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). Unfortunately the default mapping was changed, so unless you physically re-plumb the pedals, the pedal number mapping from the left is:

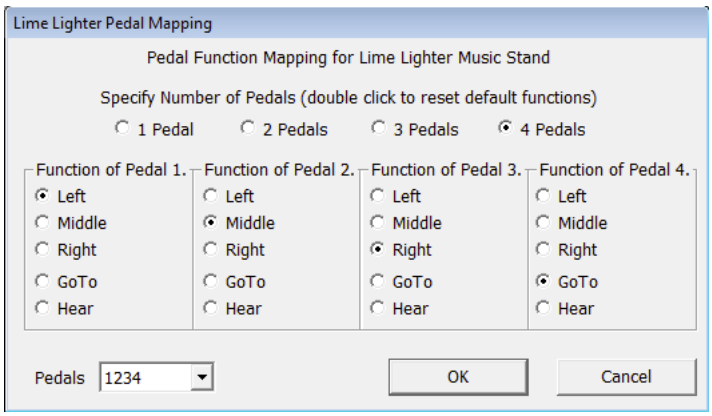
		Old BT106	New BT200 & Quad200
up arrow	F13	**1**	**1**
left arrow	F14	**2**	**3**
down arrow	F15	**3**	**2**
right arrow	F16	**4**	**4**

The pedal mapping dialog allows you to specify what the physical order of the pedals is from left to right. It also allows one to specify that you are using the old, 3-pedal *Infinity*™ device.



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'

From left to right, the physical position of a pedal on the dialog should match the actual position of the pedal on the device. This is specified using the *Pedals* ordering drop-down. When an item in the *Pedals* drop-down is selected, it changes which pedal number is assigned to each physical position. For the new AirTurn™ 4-pedal devices (e.g. BT200 & Quad200), you should specify pedals **1324**; for the old ones (e.g. BT106), you should specify **1234** (as in the dialog shown above).

The pedal ordering facility was not available in Lime 9.16. So, after upgrading to Lime 9.17, if you had a new AirTurn™ 1324 device, the apparent function assignments would be as though it were 1234. If Shift is held down when selecting a different pedal order, the actual functions are adjusted as well, so that the behaviour remains the same, but the dialog should now reflect what you see.

25.8.1.1. Old Infinity Pedal

The *Infinity*™ foot pedal, that used to be supplied with Lime Lighter, requires a DLL (*PIEHid.dll*) to drive it. On Windows, this can be installed in the same folder as Lime.exe.

The *Infinity*™ pedal can be selected using the *Pedals* drop-down. It is recommended that it is selected if you are using it. This is not essential, but will enable a warning if the DLL cannot be loaded. Unless Shift is being held down, if you select the *Infinity* pedal, the appropriate default number of pedals and their mapping will be set.

25.8.1.2. 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).

25.8.2. Lime Lighter Parameter Defaults

Using parameters, Lime has a wide range of options for modifying the appearance of a piece. Many of these are 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 computer[73]; 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.

25.8.3. Lime Lighter Arrow Keys

The old *Infinity*™ pedal unit can be used directly as the pedal uses F13, F14 and F15. However the *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 function[74].

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 Lime Lighter arrow key mode is set, then ***ctrl+alt+arrow key*** cannot be used to shove selected annotations (see section 3.22). In Lime Lighter mode, if used with ***ctrl*** and ***alt***, arrow keys always just perform their basic navigation function.

---

[73] The use of modified parameter defaults is much more efficient that the use of Parameters per se, which particularly for note heads, do reduce Lime's performance and can impact on responsiveness to pedal presses.

[74] Note that there is a Lime Lighter preference that enables one to default the arrow keys to Lime Lighter mode.

25.9. 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:

- Touch screen facilities are not available as no desktop Mac supports them.
- Zoom level limitations for wide ticker-tape contexts (see section 23.6). 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 properly erased when the highlighting rectangle is moved.
- The old *Infinity*™ pedals are not supported; only *AirTurn*™ pedals, or any pedals that can map to the arrow keys or F13 to F16, are supported. Note that, on Macs, Alt-F1 to Alt-F4 are equivalent to F13 to F16.

25.9.1. VMware Virtual Machine

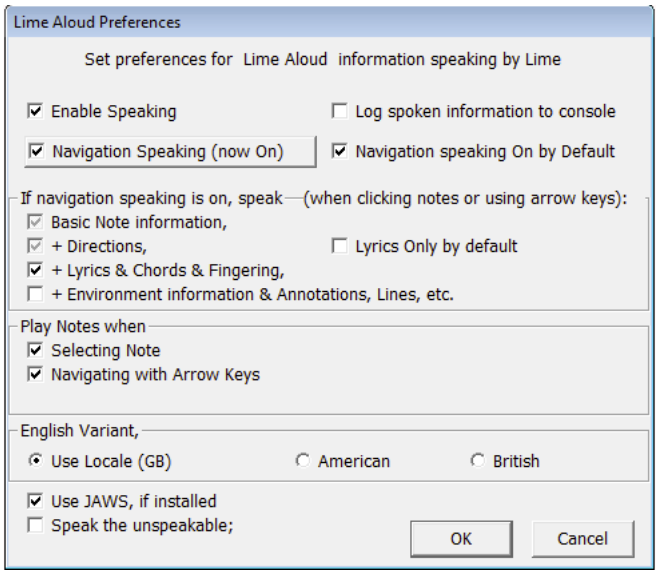
If Lime for Windows is run on a VMware virtual machine, the cursor repositioning when exiting low vision scrolling will not work properly, unless VMware's preferences have "Always Optimize Mouse for Games" set.

26. 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.

Lime aloud assumes that commas and periods will cause pauses but not be spoken explicitly.

26.1. Lime Aloud Preferences



If ***Enable Speaking*** is off, Lime itself will not speak anything. If it is enabled, then there is a choice as to whether Lime speaks about notes, etc, as one navigates through the piece; normally this will be on and will be on by default, each time Lime is started. ***Log Spoken Information to Console*** can occasionally be useful for sighted helpers to know what is spoken (for them, the speech is often too fast); this can be on, even if speaking is off.

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):

1. ***Basic Note information*** describes the selected note (notated pitch, articulation, ornaments, etc), and other information such as position (bar, beat, etc). This is always spoken.
2. Directions. All annotations that provide musical direction, including dynamics, tempo, pedals, pauses, mood, etc. Crescendo and Diminuendo hairpins are included at this level.
3. Lyrics & Chords & Fingering and Rehearsal marks. In addition to directions, any lyrics, chords, fingering or rehearsal mark annotations are also spoken. **This is the default for new users**.
4. Environment information and Annotations, Lines, etc. In addition to the first two levels, all other visible annotations (such as titles, copyright, etc), including lines and curves will be described.

The options as to whether notes are played when selected and/or when navigated to with arrow keys are the same as the equivalent in general preferences.

If ***Lyrics Only by default*** is checked then by default, when navigating, only lyric annotations will be spoken, alongside the associated note being sounded. If partially checked, basic note information will also be spoken. If checked, it overrules Lime-Aloud's normal verbosity preferences. This 'laconic' option can be temporarily turned on and off during a Lime session, so it may be better to leave the default off, and toggle it as required (see section 26.2.1).

The ***English Variant*** defines what the terms used for various musical concepts are used, if English is used. There are some significant normal-use differences between British and American English. For example a '*Crotchet*' in English is a '*Quarter*' note in American and a '*Minim*' is a '*Half*' note.

Lime Aloud normally uses JAWS. However for experimental purposes it is possible to use the Windows' built-in SAPI for basic speaking. It must be recognized that if JAWS is not used, only a subset of information will be spoken, dependent on what, if any, screen reader is active. For this, there is an option to ***Speak Extra Dialog Information***; if turned on Lime will announce dialogs, with some minimal information about them.

Prior to 9.17, Lime only spoke in the local character set. If there were characters that could not be rendered, an asterisk (*****) was substituted and spoken as such. Theoretically, with Unicode, Lime can speak anything, providing the speaker can. Unfortunately, if a voice cannot understand an exotic character, it's treated like a space, so you do not even know if there was something to speak. The ***Speak the unspeakable*** option reverts to the old method. This could be useful to hear whether there actually is an annotation, when you think there is but nothing is spoken. Otherwise it's inefficient and is not recommended, so if it is only partially checked, it will only apply to the current session.

26.2. 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.

26.2.1. Lime Laconic Lyrics

A preference option, ***Lime Laconic Lyrics***… (alt+ E P Y) invokes a simple dialog, which allows one to toggle the ***Lyrics Only*** option. Any change made will only be for the current session.

26.3. Hearing Status

In normal editing mode, the bottom row of alphabetic keys (**Z-M**) on the keyboard, 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.

Note that, of course, when editing an annotation, none of the simple keyboard keys have special functions, because they are needed for the editing.

26.3.1. Hear Annotations

Normally when an annotation is spoken, Lime tries to say what can be seen. However when the annotation uses special symbols, there can be cases where the meaning of the combination is not obvious. Lime knows about a few special cases, but it must be recognized that it cannot make a special case for every possible combination.

If an annotation is selected, the alphabetic **x** key (lower case) will cause the annotation to be spoken. If **x** is used twice in quick succession (within ½ second), if the annotation is a recognized combination, then the annotation will be spoken again in more detail, indicating its structure.

26.3.1.1. Lyric Spelling

A capital X (i.e. alphabetic **X** and shift key) can speak lyric annotations on a selected note.

- If no annotation is selected, capital **X** will read out all lyrics (if any) on the note and select the first of them.
- If a lyric annotation is selected, capital **X** will spell out the selected lyric,

26.4. 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.

26.5. 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.

The normal keyboard 'n' key has been extended as of Lime 9.17.1. If not in note entry mode and in group-select, Lime will say how many notes or annotations are selected. As of 9.18.1, if a capital 'N' (shift-n) is used and not in group-select, Lime will say the details of the currently selected note.

As of 9.18.1, if a capital 'C' (shift-c) is used, if not in group-select, Lime will say the prevailing clef, key and time signatures, and staff details. If it is a lower case 'c', Lime will just say the clef.

If a note is selected, a capital 'X' (shift-x) will cause Lime to speak any Lyric annotation on the note; furthermore, it will select the first such annotation. If a lyric is selected, a capital 'X' will spell it.

26.6. JAWS Issues

Lime Aloud for JAWS includes a set of JAWS' scripts to be used for Lime, which must be installed. At least version 9.16.6 of the scripts is needed for Lime Aloud dialogs to work properly.

26.6.1. Dialogs

With the aid of Lime's JAWS' scripts, Lime tries to make its dialogs as accessible as possible. The following are worth noting:

- For the Lime application, Lime's JAWS' scripts force 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 scripts are 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).

26.7. 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 the 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, there is an additional option (***Speak additional dialog information***, see section 26.1), which if checked, will cause Lime to speak basic information about dialogs 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 or is running in silent mode. If JAWS is active and Lime is not using JAWS, stuff can be spoken simultaneously, particularly if "*Speak additional dialog information*" is checked.

27. Lime-GoodFeel Brailling

If authorized, Lime-GoodFeel together with JAWS, can provide real-time brailling of Lime music on a braille terminal. **GoodFeel 4.2 or later is needed, to be fully compatible with Lime 9.17/9.18**.

Note that actual real-time physical brailling currently requires JAWS to drive the brailling device.

GoodFeel 3.2 does not work with Lime 9.16 or later and GoodFeel 3.5 isn't compatible with Lime 9.17. So **get an upgrade to 4.0 or later** now (**4.2** or later is strongly recommended)!

27.1. 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 Brailling. If this is unchecked, no brailling 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 brailling.

Braille Line Width. This specifies the horizontal 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 brailling does not split the braille into lines properly. If that's the case, setting it here to the width of your device is essential.

Default Save Format to: Lime Score with GoodFeel Graphics.
If this is checked, whenever the *Save As…* dialog is used the format will be defaulted to save with the graphics needed by GoodFeel. This can be convenient if one is creating scores for GoodFeel to braille.

This is not needed for real-time brailling; it is only needed for the score to be processed separately by GoodFeel (perhaps on another computer).

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 brailling.

Use GoodFeel Program, identifies whether to use the old technique of invoking the GoodFeel program itself to do the real-time brailling 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.

27.2. Braille Window

When brailling 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 in 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.

27.2.1. 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.

27.3. Brailling Timeout

Real-time brailling 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 brailling 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 brailling will be re-attempted, in the hope that the GoodFeel problem has been resolved.

Appendix A Midi Percussion Key Mapping

For MIDI Channel 10, each Midi Key number ("Note") corresponds to a different drum sound, as shown below. While many current instruments also have additional sounds above or below the range shown here, and may even have additional "kits" with variations of these sounds, only these sounds are supported by General MIDI Level 1 devices. **Middle C is C4**.

Key:	Note	Drum Sound
35	**B1**	Acoustic Bass Drum
36	**C2**	Bass Drum 1
37	**C#2**	Side Stick
38	**D2**	Acoustic Snare
39	**Eb2**	Hand Clap
40	**E2**	Electric Snare
41	**F2**	Low Floor Tom
42	**F#2**	Closed Hi Hat
43	**G2**	High Floor Tom
44	**Ab2**	Pedal Hi-Hat
45	**A2**	Low Tom
46	**Bb2**	Open Hi-Hat
47	**B2**	Low-Mid Tom
48	**C3**	Hi Mid Tom
49	**C#3**	Crash Cymbal 1
50	**D3**	High Tom
51	**Eb3**	Ride Cymbal 1
52	**E3**	Chinese Cymbal
53	**F3**	Ride Bell
54	**F#3**	Tambourine
55	**G3**	Splash Cymbal
56	**Ab3**	Cowbell
57	**A3**	Crash Cymbal 2
58	**Bb3**	Vibraslap
59	**B3**	Ride Cymbal 2
60	**C4**	Hi Bongo
61	**C#4**	Low Bongo
62	**D4**	Mute Hi Conga
63	**Eb4**	Open Hi Conga
64	**E4**	Low Conga
65	**F4**	High Timbale
66	**F#4**	Low Timbale
67	**G4**	High Agogo
68	**Ab4**	Low Agogo
69	**A4**	Cabasa
70	**Bb4**	Maracas
71	**B4**	Short Whistle
72	**C5**	Long Whistle
73	**C#5**	Short Guiro
74	**D5**	Long Guiro
75	**Eb5**	Claves [75]
76	**E5**	Hi Wood Block
77	**F5**	Low Wood Block
78	**F#5**	Mute Cuica
79	**G5**	Open Cuica
80	**Ab5**	Mute Triangle
81	**A5**	Open Triangle

[75] 75 is Lime's default pitch for its metronome

Appendix B Music Font Character Mapping

This appendix shows the mapping[75] from a numeric 8-bit character code to a music symbol in a music font. These values may be used when setting character parameters (negative values are 8-bit character codes, not 16-bit unicode).

Note that visible characters start at 32 (hex 0x20, which is a space in all fonts)[76]. In all fonts some characters have no mapping; for the SONATA font this is shown as a wee rectangle (▯); other fonts show blank.

The TUFA font only has glyphs for values below 128. MARL and SONATA have glyphs in the whole 8-bit range (32 – 255),

TUFA	-33	-34	-35	-36	-37	-38	-39
-40	-41	-42	-43	-44	-45	-46	-47
-48	-49	-50	-51	-52	-53	-54	-55
-56	-57	-58	-59	-60	-61	-62	-63
-64	-65	-66	-67	-68	-69	-70	-71
-72	-73	-74	-75	-76	-77	-78	-79
-80	-81	-82	-83	-84	-85	-86	-87
-88	-89	-90	-91	-92	-93	-94	-95
-96	-97	-98	-99	-100	-101	-102	-103
-104	-105	-106	-107	-108	-109	-110	-111
-112	-113	-114	-115	-116	-117	-118	-119
-120	-121	-122	-123	-124	-125	-126	-127

[75] The mapping tables were actually created as annotations within Lime. They are in the stationery folder of the full release in file *$Music Font Mappings.lim*

[76] Characters below 32 are 'control characters' such as carriage return, new line, or backspace.

MARL	-33	-34	-35	-36	-37	-38	-39
-40	-41	-42	-43	-44 ff	-45	-46 fff	-47 ffff
-48	-49 .	-50 _	-51	-52	-53 >	-54 ∧	-55 ∨
-56 tr	-57	-58	-59	-60	-61	-62	-63
-64	-65 .	-66 .	-67	-68	-69	-70	-71
-72	-73	-74	-75	-76	-77	-78	-79
-80 :	-81	-82	-83	-84	-85	-86	-87
-88	-89	-90	-91 .	-92	-93 ..	-94	-95
-96	-97	-98 mp	-99 pp	-100 ♮	-101	-102 ♯	-103
-104 ~	-105	-106	-107	-108	-109 f	-110 mf	-111 .
-112 ..	-113	-114	-115 ♭	-116	-117	-118 p	-119
-120 ppp	-121	-122 pppp	-123	-124 8va	-125	-126	-127
-128	-129	-130	-131	-132	-133	-134	-135
-136	-137	-138	-139	-140	-141	-142	-143
-144	-145	-146	-147	-148	-149	-150	-151
-152	-153	-154	-155	-156	-157	-158	-159
-160	-161	-162 4	-163 3	-164 6	-165 8	-166 7	-167
-168	-169	-170 2	-171	-172	-173 ¢	-174	-175
-176 5	-177	-178 ♦	-179	-180	-181 ◊	-182	-183
-184	-185	-186 ×	-187 9	-188 0	-189	-190	-191
-192	-193 1	-194	-195	-196	-197	-198	-199
-200 8vb	-201	-202	-203	-204	-205	-206	-207
-208 C	-209	-210	-211 .	-212	-213 ..	-214 ◆	-215
-216	-217	-218 .	-219	-220	-221	-222 >	-223 ∧
-224 ∨	-225	-226	-227	-228	-229	-230	-231
-232	-233	-234 ♭	-235 ♮	-236 ♯	-237	-238	-239
-240	-241	-242	-243	-244	-245 ×	-246	-247 ◊
-248 ♦	-249	-250	-251	-252	-253	-254	-255

SONATA	-33	-34	-35	-36	-37	-38	-39
-40	-41	-42	-43	-44	-45	-46	-47
-48	-49	-50	-51	-52	-53	-54	-55
-56	-57	-58	-59	-60	-61	-62	-63
-64	-65	-66	-67	-68	-69	-70	-71
-72	-73	-74	-75	-76	-77	-78	-79
-80	-81	-82	-83	-84	-85	-86	-87
-88	-89	-90	-91	-92	-93	-94	-95
-96	-97	-98	-99	-100	-101	-102	-103
-104	-105	-106	-107	-108	-109	-110	-111
-112	-113	-114	-115	-116	-117	-118	-119
-120	-121	-122	-123	-124	-125	-126	-127
-128	-129	-130	-131	-132	-133	-134	-135
-136	-137	-138	-139	-140	-141	-142	-143
-144	-145	-146	-147	-148	-149	-150	-151
-152	-153	-154	-155	-156	-157	-158	-159
-160	-161	-162	-163	-164	-165	-166	-167
-168	-169	-170	-171	-172	-173	-174	-175
-176	-177	-178	-179	-180	-181	-182	-183
-184	-185	-186	-187	-188	-189	-190	-191
-192	-193	-194	-195	-196	-197	-198	-199
-200	-201	-202	-203	-204	-205	-206	-207
-208	-209	-210	-211	-212	-213	-214	-215
-216	-217	-218	-219	-220	-221	-222	-223
-224	-225	-226	-227	-228	-229	-230	-231
-232	-233	-234	-235	-236	-237	-238	-239
-240	-241	-242	-243	-244	-245	-246	-247
-248	-249	-250	-251	-252	-253	-254	-255