



# **DORICO<sub>5</sub>**

## **Version History**

May 2023

Steinberg Media Technologies GmbH

# Contents

---

Dorico 5.0.....	3
New features .....	3
Space and Stage templates .....	3
Pitch contour emphasis .....	11
Groove Agent SE .....	13
MIDI trigger regions.....	16
Scrub playback.....	18
Instrument editor.....	19
Live note editing.....	23
Multiple item creation .....	24
Improvements.....	27
Issues resolved.....	50

# Dorico 5.0

---

23 May 2023

## New features

### Space and Stage templates

Dorico 5 helps you sculpt the sound of your playback with new tools to help you create a convincing performance: the players can be positioned on a stage, and that stage placed in a space with natural reverberation.

Increasingly, orchestral sound libraries are recorded with a variety of microphone positions and with the musicians seated in the studio or hall in their usual performance configuration, so that the ensemble has a natural stereo field and a balanced sound. If you use the tree or hall mics rather than the close mics, you also get a strong sense of the space in which the library was recorded.

This approach means that it's easy to get a convincing sound for the ensemble, because the recording engineers have done all the hard work for you. The downside of this approach is that if you want something different, it can be harder work to achieve it.

Some orchestral sound libraries – including HALion Symphonic Orchestra, the orchestra that ships with Dorico Pro – are recorded in a much more neutral way with close microphones, and with all the instruments positioned close to the center of the stereo field.

This approach provides greater flexibility, as it means the sounds can be easily used at any stereo position, and it's easy to try different levels of reverberation. But it also means that some work is required to separate the instruments on the virtual stage, and to provide appropriate effects to give the sound a particular character.

Things get more complicated still when mixing patches from different libraries, recorded in different halls or studios and with different mic configurations.

Experienced sound engineers have the skills to meet these challenges – and some improvements to the flexibility of Dorico's mixer, including the ability to create up to eight send channels, are designed to help experienced users achieve their desired results more efficiently.

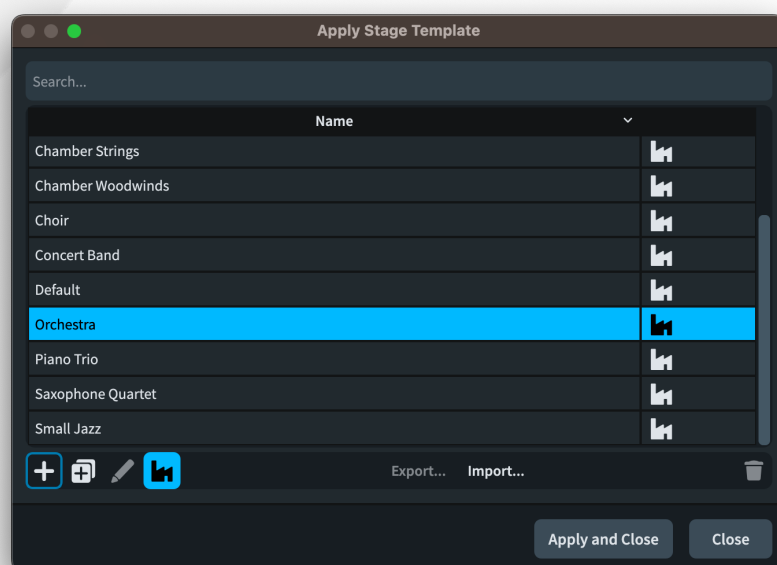
But many composers and arrangers have little experience with mixing and shaping sound in this way. For these users, Dorico's new space and stage templates will help them to produce convincing sounding ensembles more quickly and easily.

While these new features are no substitute for dedicated software solutions designed to emulate in precise detail the placement of instruments in a specific hall, such as Vienna Symphonic Library's MIR, they provide convenient ways to make greater use of the high-quality effects plug-ins supplied with Dorico, and the automatic application of appropriate settings for each type of ensemble make an immediate and positive impact on the depth of the sound of Dorico's playback, even if you are using only the factory-supplied sounds.

*Stage templates.* Stage templates describe the position of players on a nominal stage. Imagining that you are looking down on the stage from directly above, the front of the stage is at the bottom, and the rear of the stage is at the top. Players can be positioned anywhere within this space. Changes in horizontal position from left to right are translated into stereo pan values, while changes in vertical position from near to far are translated into values for send channels: typically, the counterpart space template will define two send channels, one for near instruments with less reverb, and one for far instruments with more reverb.


Stage templates can be defined very coarsely, with simple placement of entire instrument families on the stage, or very finely, with placement not only of individual instruments, but even multiple players of the same instrument, so that two violin sections can be differently placed, or four horn players, and so on.

*Applying a stage template.* When you start a new project from one of the factory templates, one of the default stage templates is automatically applied. To reapply the current stage template, which resets any adjustments you have made to the positions of players in the Live Stage, or to apply a different stage template, choose **Play ▶ Stage Template**.



Choose the template you want to apply to the project and click **Apply and Close**.

Factory templates are indicated in the dialog with a factory icon, and factory templates cannot be edited. However, you can create a new stage template based on a factory template by clicking **Duplicate Stage Template**.



*Live Stage.* To see the effect of the current stage template, open the Live Stage window by clicking  in the Mixer toolbar, either in the lower zone or the separate Mixer window.



Typically, every instrument in the project will appear on the Live Stage. However, an instrument will not appear if:

- the expression map or percussion map used for this instrument specifies that space templates should not be used, because the sound source is pre-positioned; or
- the instrument is played by a MIDI device rather than a virtual instrument.

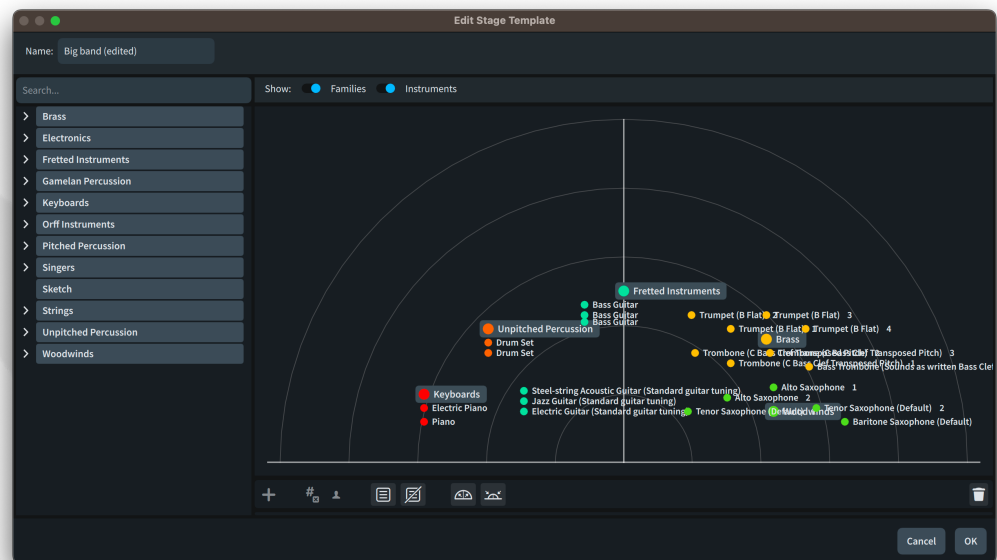
To move an instrument in the stereo field, click and drag it to the desired position. When you release the mouse button, the new position will be applied to the pan and send amounts for that instrument's output or outputs in the Mixer.

You can expand or contract the overall stereo field for all instruments at once by clicking the **Expand** () or **Contract** () buttons.

*Creating a stage template from the Live Stage.* A natural way to build your own stage template is to start from an existing project, where you can hear for yourself the effect of positioning each instrument on the stage. If you want to save the current

Live Stage setup as a stage template for future use, click **Save as new Stage Template**, which opens the **Edit Stage Template** dialog – see below.

*Editing a stage template.* To edit a stage template, choose **Play ▶ Stage Template**, choose the stage template you want to edit, and click **Edit**. If the stage template you want to edit is a factory default, click **Duplicate Stage Template** first.



**Edit Stage Template** is like the Live Stage, except that it allows you to define the positions for instruments, and entire instrument families, regardless of whether they are present in the current project.

Instrument families are shown on the stage in a larger size than specific instruments. Each family is color-coded, so both the family itself and the instruments in the family are shown in the same color.

To add an instrument family to the stage template, simply click and drag the top-level item from the tree view on the left of the dialog onto the stage. To add an individual instrument, expand the tree view to find the specific instrument you're looking for (or type its name in the search box to filter the list), then drag it onto the stage. Alternatively, simply select the family or instrument in the tree view and click the **Add** button in the action bar, which will place the instrument in the center of the stage.

Reposition any family or instrument simply by dragging it on the stage, or remove it from the stage by clicking the **Delete** button in the action bar.

As the stage becomes fuller, you might find it helpful to toggle whether the names for instruments and families appear; the switches above the stage make this

possible. You cannot select or drag items on the stage to reposition them if their names are not shown.

As you add more instruments of the same kind to the stage, they will each be numbered. You can also click the **Soloist** button in the action bar to specify where an instrument of that kind that is set to be a soloist should be placed on the stage.

You can also create groups of instruments, so that you can position those groups independently. For example, in a piece for double choir, you may want to position one choir on the left of the stage and the other on the right. The groups you define in the stage template will be used when you group players in Setup mode, as appropriate.

When Dorico is positioning an instrument added to your project according to the current stage template, it uses the following order of precedence:

- First, it will use the corresponding numbered instrument, if any;
- Second, it will use the unnumbered instrument in the group, if any;
- Third, it will use the appropriate instrument family in the group, if any.

Each of these rules is applied first to groups of instruments, if defined, and then to ungrouped instruments. If an instrument is designated as a soloist in Setup mode, then it will first prefer the corresponding soloist instrument on the stage, if any, before falling back using the order of precedence described above.

*Space templates.* Space templates describe a nominal room into which any stage can be placed. In simple terms, a space template defines one or more preset send channels in the Mixer, and a set of minimum and maximum values for pan and for the send amounts to be used for each instrument on the stage. You can define different send channels for instruments towards the front of the stage (**Near Send**) and towards the rear (**Far Send**), or use the same channel for both. (Dorico's default space templates use a single send channel with a single reverb plug-in to conserve system resources.)

As you move instruments around the Live Stage, Dorico uses the values defined in the space template to adjust the Mixer settings for that instrument's output or outputs. Lateral movement left or right adjusts the pan for that output, and the space template specifies the maximum left and right pan values that Dorico is permitted to use.

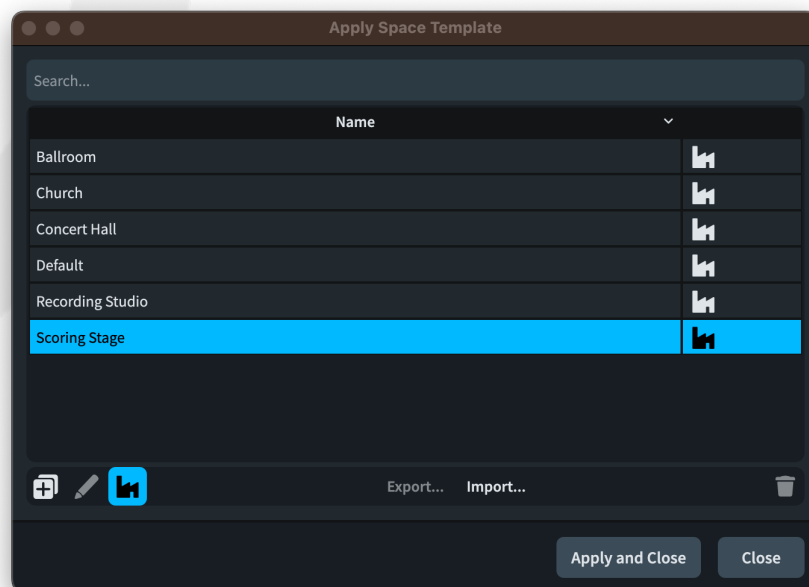
Vertical movement from near to far adjusts the send values for that output: if you have the same send channel configured for both **Near Send** and **Far Send**, Dorico simply scales the send amount between the **Near Send** and **Far Send** values as you move the instrument between the front and rear of the stage. If you have different

send channels configured for **Near Send** and **Far Send**, Dorico scales between the specified value and zero, effectively cross fading the send amount between the two sends.

*Factory stage templates.* A set of factory stage templates (**Barbershop Quartet, Big Band, Brass Band, Chamber Brass, Chamber Strings, Chamber Woodwinds, Choir, Concert Band, Orchestra, Piano Trio, Saxophone Quartet, Small Jazz**) is now included with Dorico 5. Projects created via the **Create New** page of the Hub now choose an appropriate stage template by default. Any of these stage templates can be applied to your own project via **Play ▶ Stage Template**.

Dorico includes the REVerence convolution reverb plug-in, which includes more than 90 impulse responses, providing realistic representations of concert halls, churches, cathedrals, ballrooms, and recording studios. Using REVerence, or your own preferred reverb, you can create a space with exactly the kind of sound you want to hear.


*Applying a space template.* When you start a new project from one of the factory templates, one of the default space templates is automatically applied. To reapply the current space template, which resets any adjustments you have made to the positions of players in the Live Stage, or to apply a different space template, choose **Play ▶ Space Template**.

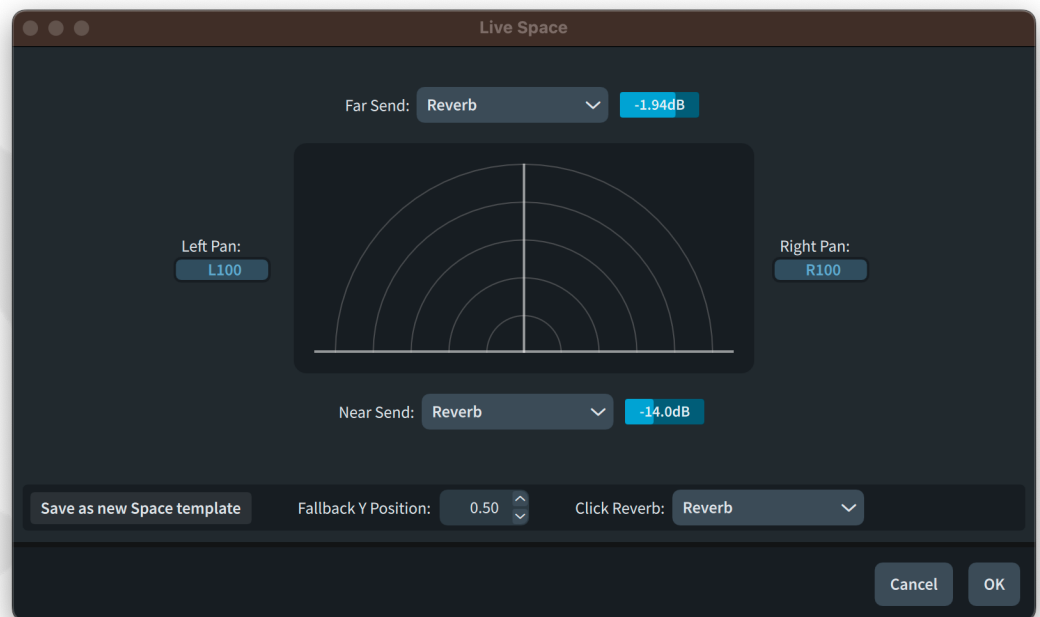


Choose the template you want to apply to the project and click **Apply and Close**.



Factory templates are indicated in the dialog with a factory icon, and factory templates cannot be edited. However, you can create a new space template based on a factory template by clicking **Duplicate Space Template**.

*Live Space.* The Live Space window allows you to adjust the maximum pan and send values for the current project. open the Live Stage window by clicking  in the Mixer toolbar, either in the lower zone or the separate Mixer window.



Configuring the Live Space is simple: the **Left Pan** and **Right Pan** values define the maximum pan value for instruments positioned at the extreme left and right of the Live Stage, and the **Near Send** and **Far Send** values define the named send channel and the amount of signal to be routed to that channel for instruments positioned at the extreme front or rear of the Live Stage. The pan and send values for instruments positioned anywhere in the two-dimensional space of the stage are determined by interpolating between these values.

**Fallback Y Position** specifies the default near-far position for an instrument that is not defined in the stage template, expressed as a value between zero (meaning the front of the stage) and one (meaning the rear).

**Click Reverb** specifies which send channel should be used for the metronome click output in the Mixer: you may want to use the same reverb you use for other instruments, a different send channel, or none.

Although you can choose which send channel in the Mixer is used for **Near Send** and **Far Send**, you cannot directly configure these channels via the Live Space window: instead, you do so in the Mixer window.

When you want to create your own space template, this is how you begin: set up the send channels in the Mixer with the plug-ins you want to use, configure the settings in the Live Space window, then click **Save as Space Template** to open the **Edit Space Template** dialog.

*Editing a space template.* Whether you choose to save a space template from the Live Space window, or you duplicate a factory space template via the **Play ▶ Space Template** dialog, you will see the **Edit Space Template** dialog. This dialog is identical in operation to the Live Space window, with the addition of an input field at the top of the dialog to allow you to specify the name of the space template.

*Multiple send channels.* You can now add FX send channels to the Mixer, up to a maximum of eight channels. Click **FX+** in the Mixer window toolbar to add a new send channel. Only the rightmost send channel can be deleted, so the **FX-** button is only enabled when selecting the rightmost send channel. You will also notice that faders for send channels in the Mixer now have a different colored background.

*Renaming Mixer channels.* It is now possible to rename channels in the Mixer. Simply double-click the channel label at the bottom of the fader, either in the lower zone Mixer or the separate window, type the new name, and hit **Return**. To reset the name back to its default, simply clear the contents of the editor and hit **Return**.

*Factory space templates.* A set of factory space templates (**Ballroom, Church, Concert Hall, Recording Studio, Scoring Stage**) is now included with Dorico 5. Projects created via the **Create New** page of the Hub now choose an appropriate space template by default. Any of these space templates can be applied to your own project via **Play ▶ Space Template**.

*Playback templates.* When exporting a playback template, if a space template is specified, it will be exported as part of the exported playback template. Subsequently importing that playback template will also import the space template, and it will appear in **Play ▶ Space Template**.

*Multiple send channels.* You can now add FX send channels to the Mixer, up to a maximum of eight channels. Click **FX+** in the Mixer window toolbar to add a new send channel. Only the rightmost send channel can be deleted, so the **FX-** button is only enabled when selecting the rightmost send channel. You will also notice that faders for send channels in the Mixer now have a different colored background.

*Renaming Mixer channels.* It is now possible to rename channels in the Mixer. Simply double-click the channel label at the bottom of the fader, either in the lower zone Mixer or the separate window, type the new name, and hit **Return**. To reset the name back to its default, simply clear the contents of the editor and hit **Return**.

*Default stage and space templates for new projects.* When you start a new project, the stage template defined in the project template will be used, if present; otherwise, the stage template specified in **Default stage template** on the **Play** page of Preferences will be used.

The choice of space template for a new project is governed by one of three things: the playback template, the choice of space template defined in the project template, or the space template override in Preferences.

If **Space template override** is set on the **Play** page of Preferences, Dorico will always use that space template when starting a new project and when applying or re-applying a playback template. If it is not set, Dorico will use the space template specified in the project template when starting a new project, or in the current playback template when it is applied.

If you have your own preferred reverb, compressor, or other settings that you always want to be applied to every project, define that setup as a space template and then choose it under **Space template override** in Preferences.

*Expression and percussion maps.* You can specify in an expression map or a percussion map whether Dorico should apply the settings of the current stage and space templates to the endpoints using these maps.

If the instrument or library you are using was recorded in situ, you probably don't want to apply the stage template, and if the library has its own recorded reverberation, you likewise probably don't want to apply the space template. For example, NotePerformer, which handles stereo position and reverb on its own, is set by default not to apply either a space or a stage template.

## **Pitch contour emphasis**

In Dorico 5, melodies play back more expressively and with a human touch, thanks to its new pitch contour emphasis algorithms. Previous versions of Dorico already include features to prevent playback sounding mechanical, including “humanization” which involves adding tiny amounts of randomness in timing and dynamics. Pitch contour emphasis augments these existing features with something more nuanced.

Human musicians naturally make higher notes in phrases slightly louder than lower ones. However, it is insufficient to simply make all notes progressively a little louder as their pitch rises: human musicians are much more subtle, and the frame of reference for changes of dynamic tends to be a time window of a few seconds.

Dorico emulates the subtle changes in dynamic in musical phrases produced by human musicians, making the performance more expressive but without exaggerating the rise and fall of the melody.

The identification of pitch contour within a melody is based on a several heuristics including relative and absolute pitch, relative note timing and absolute note duration. The effect acts each staff individually, so notes on different staves – even different staves belonging to the same instrument – will not be treated as part of the same melodic line, except for notes that have been crossed from one staff to another. For polyphonic instruments, the top note of each staff is what will be considered when identifying melodic direction.

Pitch contour emphasis is enabled by default in new projects, but is disabled in existing projects to avoid any unexpected changes in dynamics when playing back in Dorico 5.

The options for pitch contour emphasis are found on the **Dynamics** page of Playback Options, in the **Humanize** section:

- **Use pitch contour emphasis** controls whether the effect is on at all.
- **Scale effect by** allows the effect to be turned up or down relative to the default level.
- **Maximum change to written dynamic** specifies the maximum change in dynamic level that Dorico can make in either direction. However high you turn up **Scale effect by**, the change in dynamic level will never be allowed to exceed this value. The default limit is 1.5 dynamic levels, although when **Scale effect by** is 100%, the maximum effect will be closer to 1.25 levels in any case.

These values can also be overridden within an individual expression map.

Be aware that pitch contour emphasis can cause the dynamic level to go below as well as above the default dynamic level - this is necessary to produce smooth transitions between ascending and descending melodic lines. However, any decrease caused by a descending line will be much smaller than an increase caused by a rising line.

*Beat stress.* Because pitch contour emphasis produces a similar amount of dynamic change to beat stress and dynamic humanization, the existing defaults for those features have been adjusted for new projects. If you have your own custom settings, you may wish to change them.

Because pitch contour emphasis does not affect unpitched percussion, the existing beat stress options can usefully be set to a higher value for percussion than for

pitched material. As such, the **Beat stress** options in the **Note Dynamics** section of the **Dynamics** page of Playback Options now provide separate values for pitched and unpitched instruments.

Similarly, the **Humanize written dynamic by n%** option in the **Humanize** section also now provides separate values for pitched and unpitched instruments.

## Groove Agent SE

Groove Agent SE is the entry-level version of Steinberg's creative drum software and is now included with Dorico 5. It includes a version of The Kit, an acoustic drum kit recorded in Berlin's Teldex Studios, including its classic reverb chamber, and provides a choice of two mic positions.

Not only can Groove Agent SE be used as a high-quality drum kit playback device for your Dorico projects, but it also includes more than 400 MIDI drum patterns, performed by top studio drummers, and mixed by Steinberg's expert engineers. You can trigger these in Dorico using the new MIDI trigger region or drag and drop any pattern onto the track overview in Play mode to import that pattern.

Installing Groove Agent SE is optional but recommended.

*Factory playback templates.* If Groove Agent SE is installed, Dorico will show two additional playback templates: **HSSE+HSO+GASE (Pro)** (Dorico Pro only) and **HSSE+GASE (Elements)** (Dorico Pro and Dorico Elements only). These playback templates will automatically load Groove Agent SE when you add a drum set to your project and use the sounds of the included Studio Kit for playback; these sounds are of superior quality, with greater variety, than the standard drum set in the HALion Sonic factory content.


When Groove Agent SE is loaded, Dorico automatically loads the Studio Kit and a patch containing rock/country patterns. It is also configured such that patterns only respond to notes on MIDI channel 2, to avoid interference between the pitches used to trigger patterns and some of the extended techniques supported by the Studio Kit. When you use a MIDI trigger region to play patterns in Groove Agent SE, Dorico automatically directs the MIDI notes to channel 2 as required.

None of the kits included with Groove Agent SE include woodblocks, so if you add a full drum set instrument to your project, or configure the drum set to have additional instruments, Dorico will load the standard HALion Sonic drum set or other unpitched percussion sounds as available to play back those instruments.


When you first run Dorico 5, if Groove Agent SE is installed, Dorico will choose the appropriate Groove Agent-enabled playback template as your default playback template. If you are updating from Dorico 4 or earlier, Dorico will only choose the



Groove Agent-enabled playback template if the equivalent default playback template was chosen as the default in your previous version. You can adjust this at any time on the **Play** page of Preferences.

*Setting up Groove Agent SE manually.* If you are not using a playback template that includes Groove Agent SE, you will need to set it up manually in your project. The first step is to add Groove Agent SE to the rack:

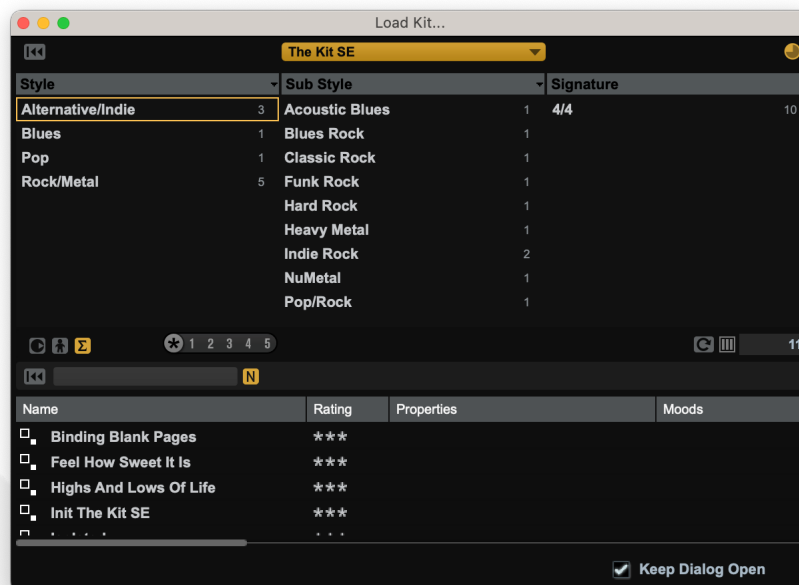
- Switch to Play mode.
- In the left zone, choose the **VST and MIDI** panel.
- Click  in the action bar at the bottom of the **VST Instruments** section to add an empty entry to the rack.
- Using the menu in the empty rack entry, choose **Steinberg ▶ Drum ▶ Groove Agent SE** to load the plug-in.

The Groove Agent SE window is split into four main areas:

- *Kit slot:* located in the top left of the window, this shows the name of the loaded kit, allows you to load a different kit, and change some settings, including whether any associated MIDI patterns should be loaded when the kit is loaded (.
- *Kit rack:* located in the top right of the window, this shows the loaded kit. (In the full Groove Agent plug-in, you can load up to four kits here, but Groove Agent SE allows only one.)
- *Pad section:* located in the bottom left of the window, this section contains transport controls, group buttons, and drum pads. When the pad section is in **Instrument** mode, each pad corresponds to a single instrument in the kit; when in **Pattern** mode, each pad corresponds to a different pattern.
- *Edit display:* located in the bottom right of the window, the edit display contains the **Edit**, **Mixer**, and **Options** pages.


At the top right-hand corner of the edit display, you will find two other useful buttons:  expands the window downwards to show a piano keyboard; and  expands the window rightwards to show the MediaBay panel, allowing you another way to browse and load kits.

To load a kit, either click the name of the kit in the kit slot, or the label below the empty slot in the kit rack. The small MediaBay panel appears:



Click the menu at the top of the panel to choose between the available kits, or click **All** to populate the panel with patches from all the kits. If you want to use Groove Agent SE's MIDI patterns, make sure the **Load Kit with Pattern** button (🎵) is activated in the top right-hand corner. Double-click the kit to load it.

Once a kit is loaded, you need to connect the plug-in to a drum set in your project:

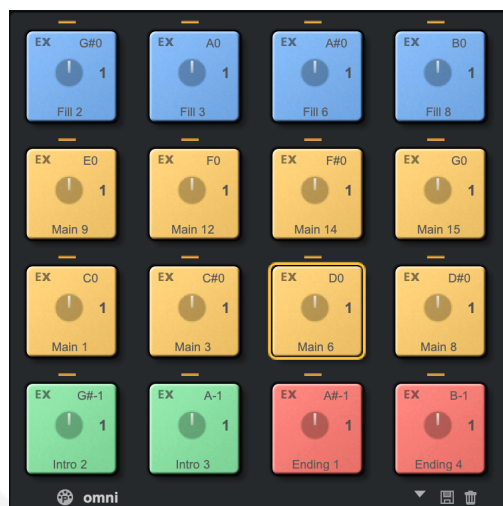
- Switch to Play mode.
- In the left zone, choose the **Track Inspector** panel.
- Select the drum set track in the track overview.
- In the **Routing** section of the track inspector, choose **Groove Agent SE** from the plug-in menu.
- Open the Endpoint Setup dialog by clicking .
- In Endpoint Setup, choose **Groove Agent SE Studio Kit** under the **Percussion Map** column for channel 1, and click **OK**.

Now when you write music in your drum set staff, it will be played back by Groove Agent SE instead of the default drum kit in your chosen playback template.

For more information on how to use Groove Agent SE, please refer to its dedicated [Operation Manual](#).

*Using MIDI patterns.* If the kit you have loaded into Groove Agent SE includes MIDI patterns, the **Pattern** button in the pad section shows a small orange activation indicator directly above it: click this, and the pad section will display a grid of patterns:





Patterns are color-coded: regular (or “main”) patterns are yellow; fills are blue; intros are green, and endings are red.

In the top right-hand corner of each pad, the MIDI pitch required to trigger this pattern is displayed. These MIDI pitches are displayed according to the convention that middle C (MIDI note 60) is C3. You may find it useful to set Dorico to use the same middle C convention, which you can do on the **General** page of **Preferences**: set **Middle C (note 60)** to **C3**.

You can drag and drop a pattern into the track overview in Play mode or into the percussion editor in the Key Editor to import that pattern into your score: a red insertion point shows where the pattern will begin.

The other way to use patterns is to trigger them using Dorico 5’s new MIDI trigger regions – see below.

## MIDI trigger regions

The Key Editor allows you to output MIDI continuous controller values, pitch bend, tempo changes, and so on, all without affecting the printed score, but one thing it cannot do is output notes that are not printed in the score.

This is what Dorico’s new MIDI trigger regions are designed to do: play a note or chord that doesn’t appear in the score for a period determined by you.

This is especially useful for triggering MIDI patterns in plug-ins like Groove Agent SE, or other pattern-based instruments such as TGuitar. If you’re writing for a real drummer or guitarist, you don’t need to write out their part in full: give them the structure of the piece, a few rhythmic cues, and chord symbols, and they’ll take care of the rest. But if you want to have some realistic playback without spending a lot of time and effort writing out music that won’t ultimately be played by your human musicians, pattern-based plug-ins can save a lot of time.



*Creating a MIDI trigger region.* You can create a MIDI trigger region as follows:

- Select the region over which you want the note or notes to be played.
- Choose **Write ► Create MIDI Trigger Region**, or use the shortcut **Shift+0** (zero), to open the MIDI trigger region popover.
- Either type the names of the notes you want to be played, or play them on your MIDI keyboard, and hit **Return**.

The MIDI trigger region is created in the score, with a light blue bar below the staff showing the names of the pitch or pitches that will be played for the duration of the region.

When the popover is open, you can repeatedly play single notes or chords on your MIDI keyboard, allowing you to audition different patterns in Groove Agent SE or another pattern-based plug-in. Each separate note or chord you play while the popover is open replaces the previous note or chord, so you can try out as many different notes as you want before you hit **Return**.

*Specifying velocity.* If you use a MIDI keyboard, you will also see the velocity of the note or chord you just played, appended to the list of pitches in parentheses. Dorico will play the notes defined in the MIDI trigger region at this velocity; if no velocity is specified, Dorico will use a default value, roughly equivalent to a dynamic of *mf*.

You can also type a velocity value when typing into the popover: simply specify the velocity as a value between 0 and 127 in parentheses following the last pitch.

*Editing MIDI trigger regions.* To change the note or chord played by an existing MIDI trigger region, select it and hit **Return** to reopen the popover: type or play the new note or chord to be played by the region, and hit **Return** to confirm the changes.

MIDI trigger regions work the same way as other regions, such as bar repeat, rhythmic slash or chord symbol visibility regions: they can be moved, lengthened and shortened using the circular handles that appear when the region is selected, or you can use the standard **Alt+←/→** shortcuts to move the region, and **Shift+Alt+←/→** to shorten and lengthen the region.

*Viewing MIDI trigger regions.* MIDI trigger regions won't print (unless the **View options** checkbox is activated in the **Annotations** section of the right zone in Print mode). They appear on-screen by default, but you can hide or show them by toggling **View ► Highlight MIDI Trigger Regions**.

*Middle C convention.* The note names displayed in MIDI trigger regions follow the **Middle C (note 60)** option on the **General** page of Preferences. If you are using a plug-in that numbers middle C as C3, like Groove Agent SE, you may find it useful to set this

option to **C3** so that the octave numbers shown in the MIDI trigger region match the octave numbers shown in the trigger pads in Groove Agent SE.

## Scrub playback

Scrubbing is a term that dates from the days of editing audio tape, where the tape was manually dragged across the replay head by rotating the reels to locate an edit point, using an action like the cleaning action of scrubbing. In modern software, scrubbing has come to mean the process of playing the audio at variable speeds, either forwards or backwards, as the user drags with the mouse or uses a jog wheel to locate an edit point.

In music notation and composition software, scrubbing can also be useful, even though the kinds of editing performed on music notation differs significantly from digital audio editing. As such, Dorico 5 introduces a new scrub playback feature that makes it easy to audition the whole ensemble or a single instrument at any point and move forwards or backwards through the playback at any speed.

*Enabling scrub playback.* You can enable scrub playback in Write mode in any of the following ways:

- Choose **Play ▶ Scrub Playback**; or
- Activate the scrub playback button on the toolbar, found between the tempo read-out and the button to open the Transport window; or
- Hold down **Ctrl+Space** (Windows) or **Alt+Space** (macOS).

When scrub playback is activated via the **Play** menu or the toolbar button, it remains active until you explicitly deactivate it, either by hitting **Esc**, choosing the **Scrub Playback** menu item again, or deactivating the toolbar button once more. However, when scrub playback is activated via the keyboard shortcut, it remains active only as long as the keys are held down.

Using the keyboard shortcut is very convenient, as it allows you to quickly activate scrub playback for just a moment. (In this way, it's like the instant print preview that can be activated by holding down the ~ key, and which temporarily hides all non-printing items while you keep the key held down.)

When scrub playback is active, as you move the mouse cursor over the music area, the cursor is shown as a speaker, and the notes that are sounding at the rhythmic position under the mouse cursor play continuously, allowing you to control when you move to the next or previous chord simply by moving the mouse cursor.

*Soloing an instrument.* By default, Dorico plays all the notes at the rhythmic position under the mouse cursor; to solo the instrument directly under the mouse cursor, hold **Shift**. If you are holding down the **Ctrl+Space** (Windows) or **Alt+Space** (macOS)

shortcut to activate scrub playback, you must additionally hold down **Shift** to solo the instrument under the mouse cursor.

*Grace notes.* Non-rhythmic notes like grace notes are not played during scrub playback.

*Repeats.* When scrubbing through music with repeats, Dorico always plays the music as if for the first pass through the repeated section.

## Instrument editor

Dorico has a database of more than 600 instrument types, which define the attributes that determine how the instrument appears, is labeled, and plays back in your project. This includes staff labels, transpositions, standard and advanced ranges, number of staves, number of staff lines, default clefs, and more besides.

Dorico 5 introduces a new **Library ► Instruments** dialog that allows you to edit these definitions and define new instruments. You can save your changes and new instruments for use in future projects.

*Opening the instrument editor.* There are two routes to open the instrument editor:

- In Setup mode, expand the card for the player holding the instrument you want to edit, open the instrument's context menu by clicking ... and choose **Edit Instrument Definition**. The instrument editor opens with the chosen instrument selected, ready for editing.
- Choose **Library ► Instruments**. The instrument corresponding to the first staff in which you have something selected in the score is selected, ready for editing.

*Project instruments and library instruments.* The instrument editor lists each instrument type only once. If a particular instrument type is in use in your project, it appears in the left-hand list with **(in score)** appended to its name. Any edits you make to that instrument definition will be applied to all the instruments of that type in the project.

For example, in the common case that you have two violins in your project, in the instrument editor you will see a single entry in the left-hand list, **Violin (in score)**, and any changes you make will be applied to both violins in the project.

If you have identical instruments in your project and you want to make changes to each of them individually, first use the instrument editor to create a new variant of the instrument definition, and then use **Change Instrument** in Setup mode to use the new variant for the relevant instrument.

For any instrument that is not yet in your project, and changes you make in the instrument editor will be saved to the project's library. When you subsequently

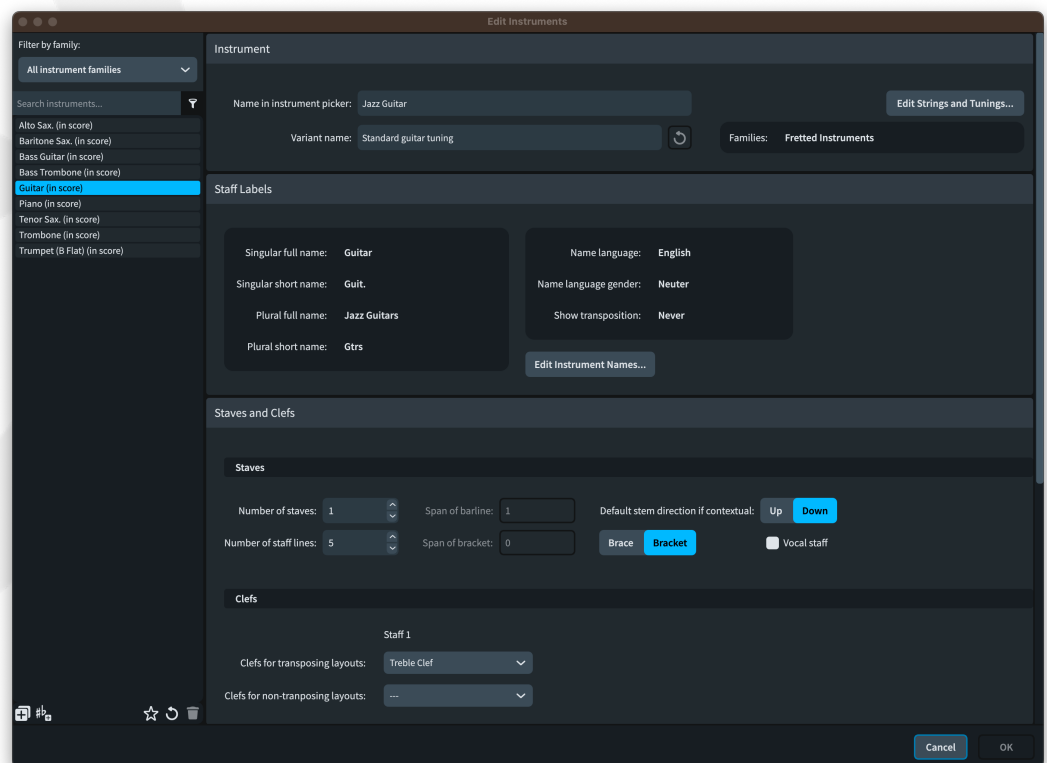
create an instrument that uses this edited instrument definition, it will reflect the edits you made earlier on.

If you want to make edits to instrument definitions, or new instrument definitions, available in future projects, click **Save as Default** (★) in the action bar: when the star icon is filled, the selected instrument definition matches the definition saved in your user library, thereafter available in all new projects.

Click the **Show only project instruments** button (🔍) at the top of the instrument list to quickly filter the list of instrument types such that only the project instruments are shown.

You can otherwise filter the list of instruments by family using the **Filter by family** menu, or by typing all or part of the instrument's name in the **Search instruments** input field at the top of the list of instruments.

*Edit Instruments dialog.* The dialog looks like this, and is split into sections, as follows:



#### Instrument section:

- **Name in instrument picker** is the name used in the middle column of the instrument picker.
- **Variant name** is the name used in the rightmost column of the instrument picker, if there are multiple variants for the parent instrument. Click the Reset variant description button to reset this to its default value.

- **Families** is a read-out showing which families an instrument belongs to. You cannot yet edit this, but this is planned for a future release.
- If the instrument is a fretted instrument, the **Edit Strings and Tunings** button will appear; when clicked, this opens the usual editor for fretted instrument strings.
- If the instrument is a percussion kit, the **Edit Percussion Kit** button will appear; when clicked, this opens the usual editor for percussion kits and instruments.

In the **Staff Labels** section, the current values for each of the names used in staff labels appears; to edit any of them, click **Edit Instrument Names**.

**Staves and Clefs** section:

- **Number of staves** determines how many staves the instrument should have by default.
- **Number of staff lines** determines the number of staff lines shown in each staff belonging to the instrument. Even if you change the number of staff lines, Dorico will still treat the instrument as a pitched instrument. Dorico does not properly support pitched instruments with staves with more or fewer than five lines, so be prepared for some interesting results if you stray too far here.
- If the instrument has more than one staff, **Span of barline** allows you to specify which staves (from top to bottom) should be joined by a barline.
- Similarly, **Span of bracket** allows you to specify which staves should be joined by a bracket or brace, and the choice of what kind of primary bracket should be used can be specified by choosing either **Bracket** or **Brace**.
- Activating **Vocal staff** means that dynamics will go above the staff by default, and that barlines that would otherwise be joined to adjacent staves in the same primary bracket group should be broken, in service of making lyrics more legible.
- For each default staff defined for the instrument, you can specify the **Clef for transposing layouts**, and optionally a different **Clef for non-transposing layout** as well.

In the **Transposition** section, you can specify the transposition interval for the instrument:

- **Written middle C (note 60) sounds as** specifies an octave transposition in concert pitch layouts: for example, assuming you are using the C4 middle C convention, for a guitar (which sounds one octave lower than written), this would be set to **C3**; for a piccolo (which sounds one octave higher than written), this would be set to **C5**.

- **In a transposing score, written middle C (note 60) sounds as** specifies the transposition in a transposing layout: for example, again assuming C4 as middle C, an alto saxophone (which sounds a major sixth lower than written) would be set to **Eb3**, while a contrabass clarinet (which sounds two octaves and one whole step lower than written) would be set to **Bb1**.
- **Uses key signature** determines whether the instrument shows key signatures, or instead shows accidentals as required for every note.

Finally, the **Range** section allows you to specify the standard and advanced ranges for the instrument, which are used when **View ▶ Note and Rest Colors ▶ Notes Out of Range** is activated. The pitches between the **End** values of **Standard range** and **Advanced range** are shown in dark red, while any pitch higher than the **End** value for **Advanced range** is shown in bright red. Likewise, the pitches between the **Start** values of **Advanced range** and **Standard range** are shown in dark red, while any pitch lower than the **Start** value for **Advanced range** is shown in bright red.

*Change instrument.* In previous versions, the **Change Instrument** command in the instrument's context menu in the Players panel in Setup mode was unavailable if the instrument had any additional staves, for example ossias, divisi staves, or extra staves added after the instrument was first created. These limitations have now been removed, though there are some caveats to be aware of:

- When reducing the number of staves in an instrument, any ossia staves associated with a staff that is removed will themselves be removed.
- When reducing the number of staves in an instrument, any extra staves beyond the new number of staves will be added as extra staves below the new lowest staff in the instrument.

It is still not possible to use **Change Instrument** with percussion kits, but we plan to address this in a future release.

*Percussion kits.* It is not currently possible to edit percussion kit instruments that exist in the current project via the **Library ▶ Instruments** dialog. As such, any percussion kits that are held by players in your project will not appear in the list on the left-hand side of the **Instruments** dialog, and you cannot choose **Edit Instrument Definition** from the context menu for the instrument in the Players panel in Setup mode. We plan to relax this restriction in a future release.

*Instrument families and score orders.* You cannot yet edit instrument families or the ordering used by Dorico when adding players and instruments to your project, but we plan to introduce editors for these in a future release.

## Live note editing

Dorico has historically never allowed the editing of the pitches or positions of notes using the mouse, with the justification that in other software this can be one of the most likely causes of inadvertent edits: you might be clicking and dragging to move the score in the view, accidentally click a note or, worse, a whole bar full of notes, and have edited the pitch before you really know what's happened.

However, we know that making these kinds of edits directly with the mouse can be very helpful, especially if you are already using the mouse when you spot the edit that you want to make. Furthermore, with the recent introduction of the Key Editor in Write mode, it became possible to edit the pitch and duration of notes in the piano roll editor using the mouse.

Dorico 5, then, extends mouse editing in Write mode from the piano roll to the main music area, allowing you to edit the pitch or duration of selected notes by clicking and dragging. We have added constraints so that you can be sure you are editing *either* the pitch *or* the duration, but never both, and there is strong visual feedback in the score to ensure that you are always aware when you are editing the music.

If you hold to Dorico's original philosophy that it should be impossible to inadvertently edit the music with the mouse, you can also of course completely disable mouse-based editing of notes in Write mode if you prefer.

*Pitch edits.* You can now repitch notes by dragging them vertically in Write mode. Hold **Alt** to copy them (so you can select a note or series of notes, hold **Alt** and drag up or down to build a chord). Dorico auditions each target pitch as you drag.

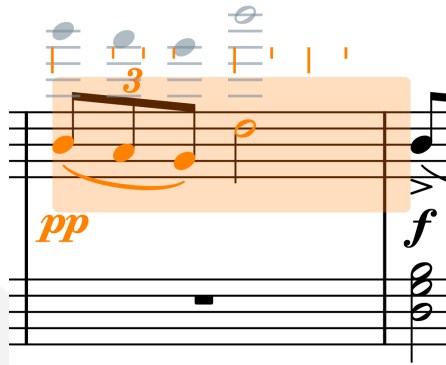
In general, dragging notes up and down transposes the notes in the same way as transposing them stepwise with **Alt**+**↑/↓**.

*Rhythmic position edits.* You can also move or copy notes to a new rhythmic position by dragging horizontally. Again, hold **Alt** to copy the music to the new position. The rhythmic grid appears as you drag to help you orientate the position at which the music will be dropped. The translucent rectangle showing the rhythmic extent of the passage being moved extends back to the start of any selected rests at the start of the selection, and to the end of the final note in the selection, to show you which music will be overwritten or displaced.

This works as you would expect with Insert mode and chord input, allowing you to, for example, swap the material in two bars by engaging Insert mode and dragging the contents of one bar to the start of the next.



*Visual feedback.* During the drag, you will see ghost noteheads that correspond to the onset position for each note, and the region that will be affected is shown by a translucent rectangle that follows the ghost notes as they move.



If you want to cancel your edit, simply hit **Esc** before you release the mouse pointer.

*Disabling mouse editing.* If you want to disable this feature, deactivate the new **Enable note pitch and rhythmic position editing using the mouse** option on the **Note Input and Editing** page of Preferences.

You can also toggle this mode of editing by assigning a custom key command to the **Toggle Mouse Note Editing** command in the **Note Input** category in the **Key Commands** page of Preferences or invoke this command using the jump bar.

## Multiple item creation

Dorico 5 overhauls item creation from a selection: you can now create multiple items at the same time, on multiple instruments, and even at different positions on each instrument.

In previous versions, selecting music across multiple staves and then creating an item via one of Dorico's popovers or Write mode panels would create a single item at the earliest selected rhythmic position on the first selected instrument. There were a couple of notable exceptions to this rule: you can create slurs and dynamics on multiple instruments at the same time, and items that modify existing notes rather than creating new items (for example, articulations and jazz articulations) can be applied to all selected notes at the same time. Additionally, during note input, when the caret is extended across multiple staves, any item you create (for example, via a popover) will be created on all staves.

Now, Dorico will create the specified item on each of the selected instruments: by default, it will do this at the rhythmic position of the first note in the selection on each instrument, which may be different, though you can specify that Dorico should instead create the item at the same rhythmic position (the earliest selected position) on each instrument if you prefer.



The items created on each instrument will also adapt to the selection on each staff. For example, when you select music across multiple instruments in which notes not only start at different positions, but the notes are of different durations or have rests at different positions, the item created on each instrument will have an appropriate duration.

Some items have further special behaviors: for example, you can now select multiple chords and create arpeggio signs on each of them at the same time; and you can create multiple glissandos at the same time, including between chords with different numbers of notes or on different staves.

*Playing techniques.* When creating playing techniques across a selection that spans multiple rhythmic positions on one or more instruments, Dorico now considers whether the underlying playback technique produced by the playing technique is a **Direction** or an **Attribute**. **Direction** playback techniques are sticky and last until they are countermanded by another playback technique, or are reset via *naturale*: for example, a pizzicato instruction. **Attribute** playback techniques, on the other hand, only last for the duration of the note on which they are written, automatically resetting at the next note; for example, an upbow or downbow instruction.

Consequently, when you create a playing technique that maps to a **Direction** playback technique, Dorico creates a single playing technique with an appropriate duration, and draws either a continuation line or repeats the symbol according to the playing technique's definition.

When you create a playing technique that maps to an **Attribute** playback technique, by contrast, Dorico creates a separate playing technique on each selected note. This makes it very quick and easy to add bowing marks: select multiple notes, for example, with **Ctrl**+click (Windows) or **Command**-click (macOS), type **Shift+P** for the playing technique popover, type **upbow**, and hit **Return**. Each selected note will have an upbow playing technique added.

*Glissandos.* In previous versions, to create a glissando you could either select both the starting and ending notes, or select the starting note and Dorico would automatically find the ending note when you create the glissando; in either case, Dorico could only create a single glissando line.

Now, Dorico will automatically create as many glissando lines as possible within the selection: if you select a series of individual notes, Dorico will create a glissando between each pair of notes; if you select a series of chords, Dorico will create as many glissandos as it can between each chord, joining the highest notes of each chord, then the next highest, and so on, until it runs out of notes in either chord; this

even works across multiple staves belonging to the same instrument, making it very quick to create sweeping arpeggios for harp.

*Arpeggio signs.* To create arpeggio signs on multiple chords at the same time, simply select each chord on which you want an arpeggio sign to appear, type **Shift+O** to open the ornaments popover, then type **arp** and hit **Return**.

*Ornaments and trills.* Dorico now creates an ornament at each note position in the selection.

A new option **Create separate trills for each selected note** has been added to the **Note Input and Editing** page of Preferences. When activated, Dorico will create a trill at each note position in the selection; when the option is deactivated, Dorico will instead create a single trill on each staff in the selection, starting at the first selected position and ending after the last selected position.

*Region-based items.* Rhythmic slash regions, repeat bar regions, numbered bar regions, chord symbol visibility regions, and MIDI trigger regions can all be created across multiple selected instruments. The position and duration of each created region will be the same on every instrument.

*Position of first created item.* By default, the position of the first created item will be determined by the position of the first note on each staff. This means you can quickly make, for example, a block selection by selecting a note or rest in one staff, then **Shift+click** in another staff, selecting notes and rests across multiple staves, then open the **Shift+D** popover and add a hairpin: the hairpin will be created on each staff, starting at the position of the first selected note on each staff.

If you would prefer Dorico to create the chosen item at the start of the selection, regardless of whether there is a note at that position on every staff, deactivate **Create items at position of first selected note** in the **Editing** section of the **Note Input and Editing** page of Preferences.

In *Behind Bars*, Gould recommends that clef changes should change at barline positions if possible, instead of immediately before the note. As such, a separate **Create clefs at position of first selected note** option allows you to specify whether clefs should be created at the start of the selection, even if you in general want items to be created immediately before the first note.

## Improvements

### Accidentals

*Improved choice of accidentals when repitching.* When using extended tonality systems, such as either of the default 24-EDO tonality systems, it is common to arrive at the desired microtonal accidental by inputting the nearest 12-EDO pitch, for example using a MIDI keyboard, and then repitch the note with **Shift+Alt+↑/↓**.

However, in previous versions of Dorico, the order in which Dorico would choose the next accidental was not always easy to predict, and nor was it guaranteed that Dorico would show all the possibilities.

Now, when repitching upwards and arriving at a new pitch class, Dorico will choose the “most flat” accidental; conversely, when repitching downwards, Dorico will choose the “most sharp” accidental. As you continue to repitch upwards or downwards, Dorico will continue to move sharper or flatter in a consistent fashion.

### Audio export

*Project activation warning.* When you try to export audio (or MIDI) from a project that is not active for playback, Dorico now shows a warning that you need to activate the project for playback and try again.

### Bar numbers

*Bar numbers coinciding with start of multi-bar rest.* If bar numbers are set to show at the start of every system, but a multi-bar rest falls at the start of the system and the option **Show ranges of bar numbers under multi-bar rests and consolidated bar repeats** is active, by default Dorico will not show a bar number at the start of that system.

However, you might prefer the single bar number to appear in its normal position at the start of the system in addition to the range shown below the multi-bar rest, in which case activate the new option **Show normal bar number if coincident with start of multi-bar rest showing range** on the **Bar Numbers** page of Layout Options.

*Horizontal alignment.* It is now possible to specify that bar numbers should be left-aligned with barlines if desired.

*Hiding bar number changes.* Bar numbers that arise from bar number changes can now be hidden via the **Hide/Show Item** command, accessed via the jump bar or a custom key command.

### Brackets and braces

*Sub-brackets for violins.* New **Sub-brackets for adjacent violin players** option on the **Brackets and Braces** page of Layout Options, allowing you to specify that adjacent

violin players, for example Violin 1 and Violin 2 section players, should use separate sub-brackets. This is especially useful if *divisi* is employed, making it immediately clear which staves belong to which player. There are three choices:

- **Single** will join Violin 1 and Violin 2 with a single sub-bracket regardless of the number of staves each shows on the system.
- **Separate** will always give separate sub-brackets for Violin 1 and Violin 2, even if each only shows a single staff on the system.
- **Separate for multiple staves** will join Violin 1 and Violin 2 with a single sub-bracket if each only shows a single staff on the system, but will otherwise only show a sub-bracket for either instrument if it shows multiple staves on the system.

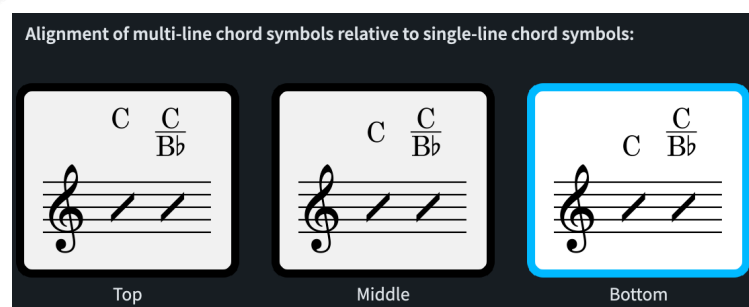
*Sub-brackets matching primary brackets.* New **When sub-bracket matches primary bracket** option on the **Brackets and Braces** page of Layout Options, allowing you to specify whether Dorico should draw a secondary bracket if the bracket will encompass the same staves as the primary bracket.

## Chord diagrams

*Used chord diagrams at start of flow.* As part of a broader set of improvements to the alignment and positioning of chord symbols and diagrams, the chord diagrams shown before the first system of the flow when **Show chord diagrams used at start of flow** is activated are now properly measured, considering the geometry of the drawn chord symbols into account as well as the chord diagrams, and ensuring that they are all contained within the bounds of the automatic frame.

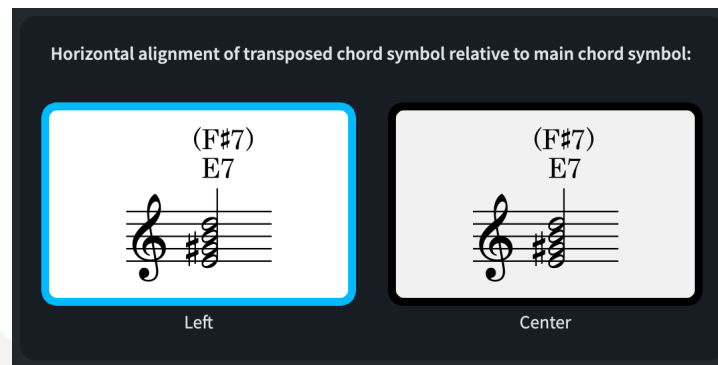
## Chord symbols

*Chord symbol alignment.* A new **Alignment of multi-line chord symbols relative to single-line chord symbols** option has been added to the **Position** section of the **Chord Symbols** page of Engraving Options.



When some chord symbols across the width of a system have altered bass notes (or are polychords) and others do not, this option determines whether the “single-level” chord symbols should be aligned with the top, middle, or bottom of the “double-level” chord symbols.

*Capo chord symbol horizontal alignment.* A new **Horizontal alignment of transposed chord symbol relative to main chord symbol** option has been added to the **Position** section of the **Chord Symbols** page of Engraving Options.



In existing projects, the upper chord symbol will continue to be left-aligned with the lower chord symbol, but in new projects the upper chord symbol will be center-aligned.

*Position of root accidentals and stacked alterations.* New options for the fine positioning and alignment of chord symbol root accidentals and the horizontal positioning of stacked alterations have been added to the **Chord Symbols** page of Engraving Options:

- In a new **Advanced Options** section under **Vertical Position** in the **Position** section, new **Vertical offset for baseline root accidentals** values allow you to adjust the vertical position of the accidental shown as part of the chord symbol root.
- A new **Vertical offset for parentheses around stacked alterations** option has also been added here, which controls the precise vertical placement of the full-height parentheses around stacked alterations.
- In the **Alterations** section, **Position of stacked alterations relative to root** determines whether vertically stacked alterations should be aligned with the chord symbol baseline (**Align bottom with root**) or center-aligned against the root (**Center align on root**).
- Also in the **Alterations** section, **Horizontal alignment of stacked alterations** allows you to choose between stacked alterations being **Left aligned** or **Center aligned** relative to each other.

*Height of parentheses.* A new **Scale parentheses to height of chord symbol with linear arrangement** option has been added to the **Parentheses** section of the **Chord Symbols** page of Engraving Options.

If a chord symbol has a linear arrangement, the parentheses can still be scaled to consider the height of, for example, the slash before an altered bass note. If you

prefer chord symbols without stacked alterations to use parentheses that match the root note height, deactivate this option. The effect is subtle, but is illustrated as follows, affecting only the first of the three chord symbols shown:

$(E/G\#)$ $(E)$ $(Em7_{\flat 5}^{\#11})$	$(E/G\#)$ $(E)$ $(Em7_{\flat 5}^{\#11})$
<i>Deactivated</i>	<i>Activated</i>

*6/9 chord symbol appearance.* A new **Use 6/9 appearance for chords with a sixth and ninth and additional alterations** checkbox has been added to the **Appearance of 6/9 chords** option in the Intervals section of the **Chord Symbols** page of Engraving Options. When this is activated, and when 6/9 chord symbols are set to display using some form of “6/9” rather than “6(add9)”, 6/9 chords with additional alterations will appear as, for example, “C6/9(#11)” instead of “C6(add 9, #11)”.

*Chord symbol regions.* Copying and pasting a chord symbol region to a staff belonging to an instrument that does not show chord symbols, or using duplicate/move to staff above or below to copy or move a chord symbol region, will now automatically update the chord symbol visibility settings for that player as appropriate.

## Clefs and octave lines

*Hiding clefs and octave lines.* Dorico has always provided a means of hiding clefs in a particular layout transposition, on the grounds that you may well want to omit clef changes if, say, the initial clef in the instrument is different; but it’s also the case that you may want some clefs to appear in the score to avoid ledger lines for the conductor, but the individual performer would be happy to read the ledger lines. Similar considerations apply to octave lines, which are in musical terms functionally the same.

As such, a new **Hide** property has been added to both the **Clefs** and **Octave Lines** groups in the Properties panel, allowing clefs and octave lines to be hidden in the current layout.

You can also use the **Hide/Show Item** command, either by assigning a custom key command or via the jump bar, to toggle the **Hide** property for clefs and octave lines.

## Condensing

*Condensing changes.* When you choose **Engrave ▶ Condensing Change** with an item other than an existing condensing change signpost selected, Dorico will now look for an existing condensing change at the position of the earliest selected item, and if it finds one, it will edit that condensing change rather than creating a new one.

## Dynamics

*Niente circle size and gap.* The default size of the *niente* circle that appears at the start or end of a hairpin denoting a gradual dynamic that either rises from or falls to silence is larger in new projects, and a new **Scale factor for niente circle** option can be found in the **Advanced Options** section of the **Gradual Dynamics** section of the **Dynamics** page of Engraving Options, allowing you to further adjust this to your taste.

Furthermore, some publishers prefer the *niente* circle to be positioned either slightly before or after the hairpin, rather than abutting it; a new **Gap between end of hairpin and niente circle** option allows you to change this as you see fit. By default, in new projects, there is now a quarter space gap between the end of the hairpin and the *niente* circle.

*Niente appearance.* A new **Style for niente 'n'** option has been added to the **Gradual Dynamics** section of the **Dynamics** page of Engraving Options, allowing you to choose between using a bold or plain italic *n* for a niente marking at the start or end of a hairpin.

*Text on hairpins.* A new **Vertical position of modifier text inside center of hairpin** option has been added to the **Gradual Dynamics** section of the **Dynamics** page of Engraving Options, allowing you to specify the default vertical offset for *poco* and *molto* text when drawn in the center of a hairpin.

*Gradual dynamic appearance.* Entering **cresc...** or **dim...** in the **Shift+D** popover will now automatically set the **Gradual style** property such that the text-based gradual dynamic will show a continuation line.

## Expression maps

*Per-switch delay.* Sounds with a slow attack or rise time can sometimes sound a little late if their notes are played exactly where written. It's common therefore to adjust the start positions of notes using these sounds to compensate for their slow attack.

You can now specify a delay in milliseconds for each switch in an expression map to accomplish this automatically: set **Delay *n* milliseconds** to the appropriate value. To make the notes sound earlier than written, specify a negative number; to make the notes sound later than written, specify a positive number.

If you have abutting notes whose playback techniques require different switches with different delay values, you may encounter some overlaps.



## Key Editor

*Velocity editor.* A spin box showing the value of the first selected note's velocity has been added to the editor header at the left-hand side. Changing the value shown here updates the velocity of all selected notes.

Furthermore, the velocity value corresponding to the vertical position of the mouse pointer within the editor is now always displayed.

*Dynamics editor.* The buttons to specify whether the selected point should be linear or constant have been added to the editor header at the left-hand side. Only points in blue regions created via the pencil or line tools can be edited in this way, so for green regions corresponding to dynamics in the score, these buttons are disabled.

*MIDI CC editor.* If **Include MIDI CC, pitch bend and dynamics from Key Editor when copying music** is activated on the **Note Input and Editing** page of Preferences, when pasting MIDI CC data, existing regions will be retained, rather than all pasted MIDI CC data being merged into a single new destination region.

*Ruler.* Double-clicking in the ruler in the Key Editor or in the track overview in Play mode now starts playback from that position.

## Layouts

*Propagate Part Formatting.* It is now possible to invoke the **Propagate Part Formatting** command from any mode via the jump bar or a key command.

## Library Manager

*Support for more library files.* The Library Manager is now able to correctly handle Dorico library files exported from other parts of the program, for example percussion kits exported from the Edit Percussion Kit dialog.

## Lyrics

*Japanese lyrics.* A new option **Use Japanese hyphens instead of extender line** has been added to the **Lyric Extenders** section of the **Lyrics** page of Engraving Options, switched off by default both in new and existing projects. When this option is activated, instead of showing a lyric extender line for syllables that should be sung over multiple notes, or for a single note written with ties, Dorico instead shows a Japanese hyphen character at each notehead position.

*Lyric extender position.* The right-hand end of the lyric extender line is now automatically aligned with the right-hand side of the notehead at the position of the end of the extender line.

*Lyric hyphens.* By default, lyric hyphens are drawn using the default paragraph style for the current line of lyrics. If the lyric syllable immediately to the left of the hyphens



uses a different paragraph style (for example, by way of the **Paragraph style** property), the hyphen now uses the same, overridden paragraph style.

*Non-breaking spaces in Edit Line of Lyrics.* In the **Edit Line of Lyrics** dialog, non-breaking spaces in lyrics appear as mid-dot characters, like this: ·. If you need to add or move a non-breaking space while editing your lyrics, you can now type **Shift+Alt+Space** to add a mid-dot that will be replaced with a non-breaking space when you confirm the dialog.

## Music fonts

*New music font families.* Thanks to the growing adoption of the Standard Music Font Layout (SMuFL), there are several music font families released under the same SIL Open Font License as Dorico's default music fonts, Bravura and Petaluma. However, installing these fonts is typically tricky, and can require either manual installation or the installation of other applications. Dorico 5 therefore now installs eight new music font families:

- **Finale Ash:** a handwritten-style font, a modern recreation of the famous AshMusic font, based on the hand of copyist Ashley Wells.
- **Finale Broadway:** a handwritten-style font, designed to emulate a thin nib ink pen.
- **Finale Jazz:** a handwritten-style font, originally created by trumpeter Rich Sigler to emulate the hand copying of Clinton Roemer, using an oblique nib ink pen.
- **Finale Maestro:** an engraved-style font, drawing on similar source material to Bravura, but with overall a less bold look.
- **Golden Age:** a handwritten-style font, a modern recreation of Don Rice's original Golden Age font, based on his own hand. The weight of the font is somewhere between the thinner appearance of Finale Broadway and the very bold appearance of Finale Jazz.
- **Leipzig:** an engraved-style font, initially developed by Etienne Darbellay and Jean-François Marti as part of the Wolfgang music notation software and today most associated with the Verovio project. It has greater stroke contrast than either Bravura or Finale Maestro.
- **Leland:** an engraved-style font, developed by Simon Smith and Martin Keary for MuseScore. It is designed to emulate the look of the venerable SCORE music notation software by the late Leland Smith.
- **Sebastian:** an engraved-style font, developed by Florian Kretlow and Ben Byram-Wigfield. Like Bravura, the overall look of Sebastian is quite bold.

In addition, several of these music fonts are paired with text font families to provide a harmonious overall look to your scores.

To try out a different music font in your project at any time, choose one from **Library ▶ Music Fonts**.

*Text font family.* If the font-specific metadata file does not specify a preferred text font family, Dorico no longer resets the font used for the default font and paragraph styles to Academico.

## MusicXML export

*System formatting.* MusicXML files exported from Dorico now specify system and frame (or, in MusicXML terms, page) breaks such that the system formatting of the layout being exported is encoded.

*Page size, staff size, and margins.* The page size and staff size specified on the **Page Setup** page of Layout Options are now encoded when exporting MusicXML. Basic page margins are also specified, though only a single set of margins is exported, so if your layout uses mirrored or different margins on facing pages, these will not be completely encoded.

*Music and text fonts.* Dorico now exports the names of the music font and main text font used in the project when exporting MusicXML.

*File extension.* Uncompressed MusicXML files exported by Dorico now have the correct **.musicxml** file extension, and report themselves as MusicXML 4.0 files.

*Noteheads.* Dorico's built-in **X and Diamond**, **Large X and Diamond**, and **Large Circled** notehead sets are now correctly exported to MusicXML.

## MusicXML import

*Holds and pauses.* Fermatas, breath marks, and caesuras are now imported from MusicXML files.

Dorico treats fermatas and caesuras as global items that affect all instruments in the system, as this reflects common notational practices, but MusicXML does not enforce this: a MusicXML file can encode a fermata, for example, on just one staff, but in Dorico, that fermata would appear at the appropriate rhythmic position on every instrument. As such, when importing these items from MusicXML, Dorico analyses the fermatas and caesuras in the MusicXML file and groups them such that it creates only the appropriate global items. This means that it is not guaranteed that fermatas and caesuras will appear at precisely the same positions as encoded in the MusicXML file, but the result should always make musical sense.

With both caesuras and breath marks, there is the additional complication that different applications specify the rhythmic position of caesuras in different ways. Dorico employs heuristics based on which application exported the MusicXML file to try to produce the intended result as often as possible.

*Ornaments.* MusicXML specifies a wide range of ornaments that can be encoded, and Dorico now imports all of these, except for delayed turns, delayed inverted turns, and shakes.

*Harmonics.* Dorico now imports natural and artificial harmonics from MusicXML files.

*Playing techniques.* A wider range of playing techniques is now imported from MusicXML files, including mute/open/stopped indications for brass instruments, and bowing marks and snap pizzicato indications for string instruments.

In addition, MusicXML allows the encoding of relatively freeform playing techniques either as text or glyphs using the **other-technical** element, and Dorico will do its best to import these as playing techniques. You may prefer to exclude these, in which case deactivate **Import other-technical playing techniques** on the **MusicXML Import** page of Preferences.

*Staff labels.* The handling of instrument transpositions in staff labels has been greatly improved. The staff label in the MusicXML file itself could be encoded as, for example, “Clarinet in Bb”, and Dorico would correctly identify this part as belonging to a Clarinet, but also import the part name from the MusicXML file exactly, then apply the prevailing layout options for staff labels, resulting in nonsensical names like “Clarinet in Bb in Bb”.

Dorico now takes a more nuanced approach, attempting to parse the part name in the MusicXML file to identify the instrument name and the transposition independently, and adjusting the layout and engraving options such that the automatic staff labels in Dorico match the appearance specified in the MusicXML file as closely as is practical.

*Voice direction for grand staff instruments.* When importing music in multiple voices, particularly for grand staff instruments like piano, Dorico’s choice of voice direction, which uses heuristics to determine whether a voice should be nominally up-stem or down-stem, did not always in previous versions conform to the default voice directions that are used when inputting notes yourself. This could lead to unexpected behavior when editing or adding more music later on.

The approach to determining voice directions has been improved such that voices with ambiguous directions now default to up-stem rather than alternating between up- and down-stem, and Dorico tries also to make sure that a single voice on a staff will always be up-stem.

## Note grouping

*Note grouping in cut time.* A new **Use common time note grouping for cut common time signatures** option has been added to the **Time Signatures With Half-Bars**

section of the **Note Grouping** page of Notation Options. When activated, this causes Dorico to group notes, beamed groups and rests in 2/2 and cut common time using the rules for 4/4 or common time.

A new property, **Use common time note grouping**, has been added to the **Time Signatures** group, allowing you to selectively apply the new notation option to group notes and rests in cut common time in the same manner as common time for an individual time signature if required.

## Note input

*Double-tap for dotted note values.* You can now specify that the next note to be input should be dotted by tapping the key for the note duration twice in quick succession: tap **6** once for a quarter note (crotchet), or tap **6** twice for a dotted quarter note (dotted crotchet).

This works both when note input is active (i.e. the caret is shown), and when editing the durations of selected notes, but not when pitch before duration input is active.

*Add bar at end of flow.* A new command **Add Bar at End of Flow** has been added to the **Note Input** category in the **Key Commands** page of Preferences, allowing you to assign a custom shortcut to add a bar at the end of the flow. This command can also be triggered via the jump bar.

*Deleting bars.* A new command **Edit ► Delete Bars** command, intended to make it easier to discover how to delete bars. Dorico determines the bars to be deleted by the positions of the earliest and latest selected items.

*Single-stem tremolos.* When joining notes in the same voice with ties, Dorico now preserves any existing single-stem tremolos on all selected staves affected by the operation, rather than only the first selected one.

## Note spacing

*Voice column handles.* When editing note spacing in Engrave mode, each global spacing column shows a square handle, and when selected, each square handle shows one or more circular handles, corresponding to the voice columns for the items in that global spacing column on that staff.

You can see at a glance when a global column has an adjustment, as the square handle will be colored red. However, to see whether a voice column has an adjustment, you had to select the corresponding square handle, as the circular handles appear only when the square handle is selected.

This has now changed in Dorico 5, such that any adjusted voice column will always show its circular handle in red. This makes it much easier to locate spacing adjustments to voice columns.

## Ornaments

*More default ornaments.* The **Baroque and Classical** section of the Ornaments panel in Write mode now includes a wider range of default ornaments.

## Page templates

*Importing page templates.* When importing page templates from library files in previous versions, it was common for any custom paragraph styles used in its text frame definitions to be duplicated, even if the destination project already contains the same styles.

Dorico is now much more selective about importing paragraph and character styles when importing page templates. A style will now only be imported if it is user-defined and does not already exist in the destination project.

*Exporting page templates.* If a page template uses one or more custom character styles in its text frame definitions, these styles are now included in the exported library file.

## Pedal lines

*Merging pedal lines.* When merging multiple selected pedal lines, Dorico now creates a retake at each position corresponding to the start of each merged pedal line.

## Playback

*Repeats playback.* Sometimes it is necessary to specify that certain notes, dynamics, or other items should be played back only on a specific pass through the music when following a repeats structure. To accommodate this, the **Suppress playback** property in the **Common** group of the Properties panel has been extended, and now allows you to specify whether the selected item should be suppressed in playback Always (on every repeat), or **On Passes** (on one or more specific repeats).

When you choose **On Passes**, by default the item will play back on every repeat. To specify on which passes the item should not play back, activate the new **Suppress on passes** property, found immediately below **Suppress playback**, and specify the passes on which the item should not play back.

The first time something is played is considered pass 1, the first repeat is considered pass 2, and so on. Specify the passes on which the item should not play back using a comma-separated list of passes, or specify a range of passes by separating the first and last pass with a hyphen. For example: **1,3,5-7**.

Be aware that suppressing playback for tempo items can have unexpected effects. Specifically, if you suppress the immediate tempo change preceding a gradual tempo change on only some passes, the gradual tempo will play back as if the

immediate tempo change were not suppressed: it will start from the suppressed tempo.

*Suppress playback.* A new **Toggle Suppress Playback** command has been added to the **Edit** category in the **Key Commands** page of Preferences, allowing you to toggle the **Suppress playback** property for the selected items using a custom keyboard shortcut, or via the jump bar.

*Pedal lines.* Pedal lines that cross repeat barlines and endings now play back correctly, considering the multiple passes through the music.

*Tremolos.* Articulations of attack (such as accents) now only affect the first played note of the tremolo, rather than being repeated on every note. By contrast, articulations of duration (such as staccato) are now applied to all the notes of the tremolo.

Measured tremolos now play back using the appropriate note length condition for the played durations of the notes in the tremolo, rather than the original written duration of the notes on which the tremolo strokes appear.

*Note length conditions.* When Dorico is determining which switch to use in the expression map, it has a system of fallbacks that it uses if the specific combination of playing techniques used in the score does not map directly onto either a single base switch, or a combination of a base switch and one or more add-on switches. If the switches in the expression map define note length conditions, Dorico will now prefer to fall back to a switch that matches, say, only one of the desired playback techniques, and additionally satisfies the note length condition.

*Dynamic power curve.* Two new options, **Minimum dynamic level** and **Maximum dynamic level**, have been added to the **Dynamic Curve** section of the **Dynamics** page of Playback Options. As their names suggest, they specify the dynamic levels at either end of the dynamic power curve, allowing you to compress the overall dynamic range.

By default, these are set to **-6** and **6** respectively, corresponding to **pppppp** and **fffff**. If your music typically doesn't use dynamics beyond **ppp** and **fff**, try setting these values to **-3** and **3**. This will allow a greater difference between these dynamic levels; any written dynamics softer or louder than the minimum and maximum defined here will play back with the minimum or maximum value as appropriate.

Used in conjunction with the existing **Dynamic curve power** option, this allows greater control over the dynamic response during playback. The new options can also be overridden in any expression map in the **Playback Options Overrides** section of **Library ▶ Expression Maps**.

*Microtonal playback.* Dorico can now play back microtonal pitches using MIDI pitch bend if required.

Using MIDI pitch bend rather than VST Note Expression (as used by e.g. HALion Sonic and Pianoteq) or VST detune (as used by NotePerformer) has a couple of notable disadvantages: because MIDI pitch bend is a channel-wide message, you cannot play chords that require different pitch bend values to produce the required pitches; and Dorico cannot determine automatically what pitch bend range is supported by a plug-in or MIDI device (it will typically be either a whole step or an octave).

Nevertheless, for certain applications using monophonic instruments, using MIDI pitch bend for microtonal playback can have advantages, too: it allows you to produce microtonal playback for plug-ins that don't support either of the other, better methods (for example, Kontakt); and MIDI pitch bend messages are included when exporting MIDI, which allows microtonal pitches to be taken into your DAW or other application. Using MIDI pitch bend for microtonal playback is opt-in, and must be activated for your plug-in via the new **Microtonality playback** menu in **Library ▶ Expression Maps**.

## Playback templates

*Overwrite warning.* When you export a playback template via **Play ▶ Playback Template**, Dorico will now warn you if the template you're exporting will overwrite an existing file.

## Play mode

*Routing for percussion kit components.* Each individual unpitched percussion instrument included in a percussion kit now appears as a separate track in the track overview in Play mode, making it possible to select each instrument and change the settings in the **Routing** section of the track inspector panel independently.

This restores functionality that was present in versions of Dorico up to 3.5 that was temporarily removed as a consequence of the rebuilding of Play mode in Dorico 4, allowing each instrument in a percussion kit to be routed to a different endpoint if necessary.

*Reset Playback Overrides.* Using **Play ▶ Reset Playback Overrides** to remove playback overrides from a large selection is now considerably faster than in earlier versions.

## Project files

*Creating new projects via the Hub.* The last value entered for **Number of bars** on the **Create New** page of the Hub is now remembered not only during the current Dorico session, but also between sessions.



## Rehearsal marks

*Horizontal alignment.* When **Position of rehearsal mark at start of system** is set to **Left-aligned with systemic barline** on the **Rehearsal Marks** page of Engraving Options, the horizontal position of the rehearsal mark is now adjusted to consider whether the systemic barline appears, so that it is always perfectly aligned with the start of the system.

## Repeat endings

*Creating repeat endings from the selection.* When Dorico uses the extent of the current selection to determine the position and duration of the first segment in a newly-created repeat ending, it now considers items including multi-bar rests.

## Rests

*Rest positioning.* Dorico has always provided the option to align rests within a beat group at the same vertical position, but not rests in different beat groups. This can lead to successive rests within a bar being assigned different vertical positions, and you might prefer them to be assigned the same position.

A new **Align rests in the same voice vertically** option has been added to the **Rest Positioning** section of the **Rests** page of Notation Options. Set this to **When in the same bar** to ensure that all rests in a voice are assigned the same vertical position.

## Scripting

*Script Console.* **Script ▶ Console** allows you to run Lua commands in an interactive fashion. The console has been given some significant quality of life improvements in this release:

- The **Script ▶ Hide Console** menu item has been removed; instead, **Script ▶ Console** is now a toggle, so you can toggle the visibility of the console with a single key command or via the jump bar.
- A new **Always on top** checkbox has been added to the console, allowing you to specify that the console should always float on top of other windows. Dorico will remember the state of this checkbox not only during the current session, but between sessions.
- The size and position of the console is now preserved between sessions.
- You can resize the upper log view and lower command entry panes in the console by dragging the splitter between them. The relative size of these panes is also preserved between sessions.
- A monospaced font, Source Code Pro, is used for the console output.
- When the console opens, the focus is put into the command entry pane by default. The command entry pane now has a distinctive appearance, with a white background and line numbers shown in the left margin, so it is now



much clearer which pane you can type into, and which shows the interpreter's output.

- The command entry pane now accepts multi-line input: simply hit **Return** to add a new line. To execute the script, either click the **Execute** button, or use the shortcut **Ctrl+Return** (Windows) or **Command-Return** (macOS).

*Recording a macro.* When you choose **Script ► End Recording Macro** to complete a macro recording, Dorico now prompts you to specify the name of the script, and automatically presents a file chooser dialog defaulting to the appropriate location in your user application data folder to save the script such that it will appear in the **Script** menu.

*Submenus for scripts.* If you create subfolders inside the **Script Plug-ins** folder in your user application data folder, Dorico uses the names of these subfolders to create submenus in the **Script** menu, making it easier to categorize your script plug-ins. Dorico also loads scripts from the global application data folder, if any are found there.

*Jump bar.* Scripts that appear in the **Script** menu can now be triggered via the jump bar: simply type the beginning of the script name, and it will appear in the list of commands that can be executed. If the script appears in a submenu of the **Script** menu, the submenu name is appended to the name of the script in parentheses.

*Scripts that sequence multiple commands.* In earlier versions, scripts that execute multiple commands in sequence would not always work as expected: for example, a script that attempts to set two different properties in succession on a single selected note would often fail, because Dorico would not notify about each edit before attempting to perform the next one. Dorico now takes a different approach to running scripts, such that each edit can notify, and as a result, scripts now run more reliably. This also means that all the edits performed by a script can now be undone in a single step.

## Setup mode

*Renumber instruments.* A new **Setup ► Renumber Instruments** command has been added, which renumbers the instruments in the project according to the order in which the players are listed in the Players panel in Setup mode.

This is useful if you have, say, four horns in your project, and after having written music for them, you want to change the order of the players: simply dragging the players into a new order in the Players panel won't renumber the instruments, but now you can do this with one click.

*Instrument transpositions.* Dorico now only appends a text description of an instrument's transposition to its name as it appears in the Players panel if the

instrument has variants that define different transpositions; hitherto Dorico has appended the transposition for instruments that have variants of any kind. For example, baritone saxophone provides variants with different clefs for concert and transposed pitch, but the actual transposition of the instrument is the same in all cases. As such, baritone saxophone used to appear in the Players panel as **Baritone Saxophone (E flat)**, which is redundant: now, it appears simply as **Baritone Saxophone**.

Relatedly, this instrument name is also used to create the default name for each part layout. This would result in the boxed text in the top left corner of the first page of each layout showing the transposition in words, for example, “Clarinet (B flat)”. Dorico now automatically uses the appropriate accidental symbol, so the boxed text instead shows, for example, “Clarinet (B $\flat$ )”.

## Staff labels

*Player names as staff labels.* A new option **Show player names** has been added to the **Staves and Systems** page of Layout Options allowing you to specify whether staff labels should use player names rather than instrument names **On all systems** or **On first system only**.

*Leading for editing staff labels.* In simple cases, overrides made to the horizontal alignment and leading of staff labels in the **Edit Instrument Names** dialog are now carried through to the resulting staff labels.

## Staff spacing

*Frame fullness indicator.* The frame fullness indicator now appears in the left page margin rather than in the right margin, to make the difference between the indicators for vertical and horizontal fullness clearer.

## Steinberg Licensing

*Latest version of Steinberg Activation Manager required.* Dorico 5 requires Steinberg Activation Manager version 1.4.30 or later. If you are running an earlier version of SAM, please ensure you update it. The easiest way to update SAM is by running Steinberg Download Assistant.

## Text

*Text frame background and border color.* Two new properties, **Background color** and **Border color**, have been added to the **Text** group in Properties, allowing you to modify the appearance of text frames. To use these properties, activate frame editing in the Engrave mode toolbox, then select the text frame whose border or background color you want to override.

*Select More.* A new special behavior has been added to **Edit ▶ Select More**: if you select a single staff- or system-attached text item and invoke **Select More**, Dorico will select only other text items that use the same paragraph style.

*Change Paragraph Style.* New **Edit ▶ Notations ▶ Text ▶ Change Paragraph Style** dialog, allowing you to change the paragraph style used for selected staff- or system-attached text items. This is handy when used in conjunction with the new ability to use **Select More** to select more text items that use the same paragraph style as the first selected text item.

*Unicode text input.* If you want to input a music symbol in a text item in Dorico, the easiest way is to right-click and choose **Insert Music Text** from the context menu, which opens a dialog showing all the music symbols in SMuFL; any symbol you insert is automatically set to use the **Music text** character style.

Until now, however, it has been awkward to input other arbitrary Unicode symbols, but Dorico 5 introduces a new way to do this: when editing text, either for a staff- or system-attached text item or in a text frame, type the four-character hexadecimal number that represents the code point of the character you want to insert, then type **Alt+X**. Dorico converts the four characters to the left of the insertion point into the appropriate Unicode character.

For example, the horizontal ellipsis character has the code point U+2026. To insert this in Dorico, type **2026** into the text editor, then type **Alt+X**.

Hexadecimal numbers use the Arabic digits 0–9 and the Roman letters A–F. For one more example, the interpunct or middle dot character has the code point U+00B7. To insert this in Dorico, type **00B7** into the text editor, then type **Alt+X**. You do need to include any leading zeroes so that the code point is always expressed as four characters.

You can also do this in reverse: if the four characters immediately to the left of the insertion point are not recognized as a hexadecimal number, Dorico will instead convert the character immediately to the left of the insertion point into its code point.

*Leading control.* It is now possible to adjust the leading (line spacing) of an individual text item via the text editor popover, using the control shown on the right.



*Tokens for project duration.* New tokens for the total duration of the project have been added, and can be used in text frames:

- **{@projectDuration@}** uses smart quotes, for example 4'33".

- `{@projectDurationStraightQuotes@}` uses straight quotes, for example 4'33".
- `{@projectDurationPrimes@}` uses primes, for example 4'33".

To insert one of these tokens, right-click while editing the text in a text frame, and choose the appropriate entry from the **Project Info** submenu.

*Font families with extended styles on macOS.* One of the troublesome differences between the way Windows and macOS handles font families concerns extended styles beyond the four standard style names (Regular, Bold, Italic, and Bold Italic). Windows only allows these four styles and will automatically synthesize a missing style – so if a font family does not provide, say, a bold weight, Windows will dynamically create one. macOS, by contrast, does not have a fixed idea about the styles that need to exist in a font family.

Take, for example, the Futura family that is included with recent versions of macOS. Although a much wider range of styles is available in alternative cuts of this typeface from Linotype and URW Type Foundry, the version that ships with macOS provides five styles: Medium, Medium Italic, Bold, Condensed Medium and Condensed ExtraBold.

On Windows, each of these weights would appear as a separate font family with the four standard styles: Futura Medium Regular, Futura Medium Bold, Futura Medium Italic, Futura Medium Bold Italic; of these, only Futura Medium Regular is an actual font file on disk; the other three are synthesized by Windows itself. Futura Medium Bold, for example, would not use the actual Futura Bold font file, because as far as Windows is concerned, “Futura” and “Futura Medium” are distinct families.

On macOS, by contrast, styles are never synthesized by the operating system: in most macOS applications you can only choose the styles that are really installed.

This presents some difficulties for cross-platform applications, since beyond the fundamental difference in how extended styles are handled, there are other difficulties, such as font files defining family and style names in multiple ways, and different operating systems and even applications on those operating systems electing to use different information to find and name fonts.

Previous versions of Dorico have had an unhappy compromise in trying to handle these platform differences. On Windows, everything is straightforward: all font families have four standard styles, and beyond that, any further styles appear as distinct families, as the operating system intends. But on macOS, Dorico has shown Windows’s four standard styles in addition to the actual named styles provided by the font family, and relied on the font substitution behavior of the underlying

application framework to choose an appropriate style if one of the standard styles is unavailable.

To return to the example of the Futura font family included with macOS, it was possible in previous versions of Dorico to choose **Futura** as the font family for a font, paragraph or character style and leave the style set to **Regular**, despite this named style not existing. In practice, for text displayed in your project, Futura Medium ends up being used.

However, when that project is subsequently reopened, Dorico will warn that Futura Regular is missing. Because you can see text using Futura in the project, you might consider the warning spurious, but in fact it is correct: Futura Regular is being requested, but it is not available.

To address this, Dorico on macOS now only shows the named styles for a font family that are present on the system. This affects everywhere that font family and style names are shown in the application, including the Font Styles, Paragraph Styles and Character Styles dialogs, and the text editor popover.

When you change the font family to one that does not implement the currently chosen named style, Dorico explicitly chooses the first available named style. If you choose Futura, for example, Dorico will automatically choose the Medium style, rather than making it appear that it has chosen the Regular style.

Furthermore, Dorico now also automatically updates all font and paragraph styles that define the edited style as their parent and which inherit its choice of font family, to ensure that every dependent style also specifies a named style that exists.

This means that when Dorico will not allow you to choose an unavailable style by default, and you will not see any warnings about missing fonts when you reopen the project.

Unfortunately, this does nothing to address the inevitable difficulties that can occur when moving projects that use these kinds of extended font styles between macOS and Windows. It is still necessary to carefully remap the fonts via the Missing Fonts dialog when opening such a project on the other operating system.

## Trills

*Double sharp accidentals.* In previous versions, the size of a double sharp accidental above a trill marking indicating the trill interval was too small to be read comfortably; Dorico now enlarges the accidental to make it clearer.

## User interface

*Follow playhead during playback.* A new **Follow playhead** slide switch has been added to the status bar. By default, when starting a new project, this follows the **Follow**

**playhead during playback** option on the **Play** page of Preferences. You can determine whether Dorico should scroll the score, Key Editor and track overview in Play mode to follow the playhead during playback independently for each project window. This can be useful if you have multiple windows open on the same project, and want one to follow the playhead during playback, and another to remain static unless you explicitly move the view. This setting is preserved between sessions for each window.

If you prefer to toggle this setting via a custom key command, you can assign a shortcut to **Window ► Toggle Follow Playhead** in the **Key Commands** page of Preferences, or invoke this command using the jump bar.

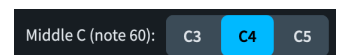
*Layout transposition.* The read-out that shows whether the current layout is shown in **Concert Pitch** or **Transposed Pitch** has been replaced with a pair of toggle buttons, allowing you to change the current transposition directly without using the **Edit** menu.

*View type.* The drop-down menu for choosing between **Page View** and **Galley View** has been replaced with a pair of toggle buttons, saving you a click when changing the view type with the mouse.

*Middle C convention.* Different hardware and software manufacturers use different numbering schemes for octaves, which can lead to confusion when trying to cross-reference between different devices, software applications, and plug-ins.

Dorico by default numbers middle C (MIDI note 60) as C4, following the widely-used International Pitch Notation standard, which is also the standard followed by manufacturers such as Roland for their synthesizer products. However, Steinberg's other host applications, including Cubase, and plug-ins, including HALion, by default number middle C as C3. Yamaha also numbers middle C as C3 in their synthesizer products. A third scheme, less widely used but popularized by Cakewalk's products, numbers middle C as C5, so that MIDI note 0 is numbered as C0, rather than as either C-1 (when middle C is numbered as C4) or C-2 (when middle C is numbered as C3).

Dorico now allows you to choose how you want middle C to be numbered throughout the application. To change this setting, open the Preferences dialog, and on the **General** page, set **Middle C (note 60)** to **C3**, **C4**, or **C5**, as you prefer.



This setting affects octave numbering throughout the application, including: the piano roll in the Key Editor; the Keyboard panel; the selected note read-out in the status bar; MIDI trigger regions; the **String** property in the Properties panel; the **Edit**

**Strings and Tuning, Transpose, Filter Notes by Pitch** and **Write ▶ Transform** dialogs; and everywhere where you can specify a pitch using a spin box.

One place in the application that has an independent representation of middle C is **Library ▶ Expression Maps**: this allows you to match the numbering scheme used in the documentation of a third-party plug-in or sample library manufacturer without changing the representation of middle C everywhere else in Dorico.

*Instrument transpositions.* Instruments that use a different clef depending on whether the layout uses concert or transposed pitch are now described in the instrument picker in Setup mode in these terms, instead of misleadingly being described as using one clef in full score layouts and another in part layouts.

*Tool tips.* All buttons in the project window now once again show tool tips if appropriate.

*Missing fonts.* When Dorico shows the **Missing Fonts** dialog during project opening to report that text items, font styles, paragraph styles and character styles in the project use fonts that are not installed on the system, the dialog now additionally shows the name of each style that specifies the missing font.

*Transpose dialog.* When returning to the **Transpose** dialog during the same editing session, Dorico now remembers the last chosen value for **Apply atonal spelling options**.

*Audio export dialog.* When returning to the **Export Audio** dialog, Dorico now remembers the last chosen audio format and (if applicable) bit depth.

*Localization.* When running Dorico in Spanish, Portuguese, and Chinese, some default translations (for example, for buttons like **OK**, **Cancel**, **Apply**, **Close**, etc. in dialogs) were missing; these default translations are all now included.

*Background colors.* It is now possible to define gradients with any number of stops in the **gradients.xml** file in the user-level application data folder. Dorico will display these gradients as intended in the music area background, though gradients with more than four stops, or where the distribution of the stops is irregular, will not appear as intended in the **Colors** page of Preferences.

*Device Setup.* If Dorico detects that the current audio device is only presenting a single output, rather than a stereo pair, a warning now appears in **Edit ▶ Device Setup**, together with a link you can follow to a knowledge base article with instructions for reconfiguring your audio settings in Windows to solve the problem.

*Default plug-in output slider.* The **Default output level** slider on the **Play** page of Preferences now shows the effective gain level of the current slider value.



*Missing playback components.* During start-up, Dorico checks whether the recommended VST instruments and sound content are correctly installed. If it determines that any component or content is missing, the message it shows now tells you the exact name of the download choice you need to download and install via Steinberg Download Assistant.

If you prefer not to use the instruments and content included with Dorico and don't want to install them, you can activate the **Don't say this again** checkbox in the prompt to suppress the warning in future. However, certain components are installed by the main Dorico application installer itself, and their absence could indicate a wider installation issue, so this message cannot be suppressed: to prevent this message from appearing in future, please reinstall Dorico via its application installer to repair your installation.

*Automatic crash report collection.* In the unlikely event that Dorico crashes, the next time you start the application, you will be shown a prompt to send the crash report to Steinberg. Crash reports do not contain personally identifiable information and crash logs are automatically renamed when they are uploaded to our servers so that even the original filenames are changed.

If you prefer, you can completely opt out of automatic crash report collection, and in any case Dorico will never send crash reports to Steinberg without first prompting you: every time it finds a crash report, it will show you a prompt and ask if you want to send the log to Steinberg.

If you choose to opt out and later want to opt back in, activate **Show prompt to send crash reports to Steinberg** on the **General** page of Preferences.

We encourage you to send crash logs to Steinberg. They allow us to find and fix problems in our applications and deliver improved software quality to you and all Dorico users.

*Guided tour in multi-user environments.* Dorico will now read the **finishedTours.xml** file that records the completion state of the guided tour in each of the project window's modes from the global user data folder, if not found in the user's own data folder. This is useful in multi-user environments such as schools, colleges, and universities.

## Improvements in Dorico Elements

The capabilities of Dorico Elements have been expanded, as follows:

- Projects can now have an unlimited number of players (previously limited to 24 players).
- **Play ▶ Playback Options** is now included.



- **Font Styles, Paragraph Styles and Character Styles** are all now included in the **Library** menu.
- Note spacing and staff spacing tools are now included in Engrave mode.

Dorico Elements also gains all the new features added in Dorico 5, except for the new **Library ▶ Instruments** dialog.

The above new capabilities are also available in Dorico for iPad, with an active monthly or annual subscription, or if the Lifetime Unlock in-app purchase has been purchased. Dorico for iPad with an active subscription or Lifetime Unlock also gains support for playback through an external MIDI device.

### **Improvements in Dorico SE**

Dorico SE's capabilities have also been expanded, and you can now work on projects with up to eight players. (As before, projects with more than eight players will open read-only, allowing playback and printing.)

Dorico SE also gains many of the new features added in Dorico 5, except for **Library ▶ Instruments**, stage and space templates, scrub playback, MIDI trigger regions, and Groove Agent SE.

Dorico for iPad is similarly expanded, with the free version also now able to work on projects with up to eight players (12 after you sign in with your Steinberg ID).

## Issues resolved

Component	Issue
<i>Accidentals</i>	Under some circumstances, changing the accidental of a note would not cause the accidentals on notes within the same bar to be recalculated right away as required; this has now been fixed.
<i>Articulations</i>	Articulations that consist of multiple symbols would draw incorrectly when positioned on the stem side of the note; this has now been fixed.
<i>Bar numbers</i>	Under some circumstances, bar numbers would not update correctly when undoing changes to local time signatures; this has now been fixed.
<i>Brackets and braces</i>	Dorico will no longer allow a secondary bracket to join adjacent players with identical instruments that are in different player groups.
<i>Chord diagrams</i>	When changing the tuning used for chord diagrams shown for a player, Dorico no longer unnecessarily resets the choice of shape back to the default one for that tuning.
<i>Chord diagrams</i>	Chord diagrams that are created via <b>Generate Chord Symbols from Selection</b> will now show fingerings if enabled via Layout Options.
<i>Chord symbols</i>	When changing the capo transposition for chord symbols, Dorico now automatically resets the appearance of any chord symbols that have been overridden; this prevents overridden chord symbols from continuing to draw using the root of the original capo transposition.
<i>Chord symbols</i>	Under some circumstances, it was possible for saved chord symbol appearances in <b>Library ▶ Chord Symbols</b> to have a mismatch between their saved transposition and the displayed chord symbol, resulting in chord symbols showing the wrong root in the score; this has now been fixed.
<i>Chord symbols</i>	Under some circumstances, Dorico could crash when using <b>Edit ▶ Reset Appearance</b> in a layout showing both chord symbols and chord diagrams; this has now been fixed.
<i>Clefs</i>	When <b>Clef at start of systems following first system</b> on the <b>Clefs</b> page of Notation Options is set to <b>Hide clefs</b> , Dorico now only hides the clef if the system contains a single instrument, and if that single instrument shows only a single staff.
<i>Comments</i>	It is no longer sometimes possible for Dorico to get caught in a loop of selecting and reselecting the same two comments in the Comments panel when navigating quickly through the list of comments with the arrow keys.
<i>Dynamics</i>	In rare circumstances, grouping dynamics, then undoing, then re-grouping the same dynamics would fail; this has now been fixed.
<i>Dynamics</i>	Deleting individual dynamics from different groups of dynamics now works correctly, instead of only deleting the selected dynamics from the first group.

Component	Issue
<i>Grace notes</i>	Under rare circumstances, grace notes could unexpectedly not appear at the start of a system where an explicit system break is present; this has now been fixed.
<i>Guitar notation</i>	When the <b>Pre-bend interval</b> property is set to a negative value, the automatic allocation of notes to strings and the <b>String</b> property could behave incorrectly, causing some valid pitches to display incorrectly as ?; this has now been fixed.
<i>Holds and pauses</i>	When a fermata appears both above and below a rest, the fermata below may have been incorrectly positioned if the single drawn rest represents multiple voices; this is now fixed.
<i>Jump bar</i>	It is now possible to redefine the shortcuts used to switch between <b>Commands</b> and <b>Go To</b> modes in the jump bar.
<i>Keyboard panel</i>	When dragging the range selector above the main keyboard to either extreme of the range, when the mouse pointer leaves the range selector during the drag, it was possible for the drag to be unexpectedly re-interpreted as changing the displayed range; this has now been fixed.
<i>Key Editor</i>	Dorico could crash when updating the Key Editor when selecting something in the music area resulted not only in the piano roll being replaced by the percussion editor but also switching to a different flow; this has now been fixed.
<i>Library Manager</i>	Under some circumstances, opening a new project or project window with the Library Manager visible could cause the Library Manager window to become blank until you start interacting with it again; this has now been fixed.
<i>Lines</i>	Lines drawn with repeatable symbols now print in the correct color if their color has been overridden via the <b>Color</b> property.
<i>Lines</i>	The <b>Start position</b> and <b>End position</b> properties now once again appear correctly in the Properties panel for rhythmic position-attached lines shown within the staff.
<i>Localization</i>	After changing the user interface language via Preferences, all key commands now work as expected, according to the appropriate keyboard language for the newly-chosen language.
<i>Localization</i>	Toggle buttons in some dialogs were not correctly translated into the appropriate language when running Dorico in a language other than English; this has now been fixed.
<i>Lyrics</i>	When multi-pasting music containing lyrics to several instruments, lyrics could be incorrectly linked across multiple staves; this has now been fixed.
<i>Lyrics</i>	Lyrics with formatting changes within them are no longer sometimes incorrectly positioned as if they had hanging punctuation.
<i>Music fonts</i>	If a music font specifies a particular text font in its metadata file, Dorico now correctly applies the chosen text font to the default font and paragraph styles.

Component	Issue
<i>MusicXML export</i>	Dorico no longer sometimes incorrectly exports notes that are not visible in the layout (for example, because they are covered by a rhythmic slash region) without specifying their pitch, which is invalid MusicXML; instead, such notes are simply omitted.
<i>Note input</i>	When determining which endpoint to use for echoing notes played on the MIDI keyboard, Dorico now considers independent voice playback, and chooses the most appropriate endpoint based on the selection, rather than simply always choosing the first endpoint for the instrument.
<i>Note input</i>	It is no longer possible to delete all the material in a flow in Insert mode such that the flow ends up with zero duration.
<i>Note input</i>	Input via MIDI keyboard into a percussion kit or an individual unpitched percussion instrument that has been imported from a library file now works reliably; in previous versions, Dorico could fail to choose the expected playing technique.
<i>Note input</i>	Under certain circumstances, removing material via Insert mode or the <b>Shift+B</b> popover could fail to remove tuplets correctly, for example when the notes in the tuplet to be removed are slurred to notes outside the tuplet; this has now been fixed.
<i>Note input</i>	It is no longer possible to add a rhythm dot to a selected implicit bar rest.
<i>Note input</i>	Under some circumstances in transposed layouts, Dorico could add intervals with the wrong note spelling when using the <b>Shift+I</b> popover; this has now been fixed.
<i>Note input</i>	Under some circumstances, attempting to navigate from the first or last rest in a voice in a percussion kit could cause a crash; this has now been fixed.
<i>Play mode</i>	It is once again possible to duplicate the VST instrument selected in the <b>VST and MIDI</b> panel in Play mode with a key command or via the jump bar.
<i>Playback</i>	The value of <b>Increase written dynamic of other beats in the bar</b> on the <b>Dynamics</b> page of Playback Options is no longer <i>subtracted</i> from off-beat notes.
<i>Playback</i>	When applying the <b>Min./Max. velocity</b> values specified in expression map switches, Dorico now correctly considers any defined note length conditions.
<i>Playback</i>	If you have an immediate dynamic and a gradual dynamic at the same rhythmic position and they are not in the same group, the immediate dynamic will now be processed for playback before the gradual dynamic, so the dynamic change will be as expected.
<i>Playback</i>	When editing linked slurs or dynamics that are linked to instruments not shown in the current layout, when you subsequently play back, those instruments not present in the layout no longer unexpectedly play back.

Component	Issue
<i>Playback</i>	A <i>mesa di voce</i> hairpin of the form <i>diminuendo-crescendo</i> no longer plays back in the opposite direction.
<i>Print mode</i>	When toggling between <b>Mono</b> and <b>Color</b> , the <b>Watermark</b> checkbox is now always correctly enabled or disabled as appropriate.
<i>Project files</i>	When starting a project from a user project template, if a new flow needs to be created, it is now created before playback is set up, thus ensuring that playback will always work as expected right away.
<i>Project files</i>	When starting a new project via the Hub, the <b>Project will use multiple flows</b> setting will now apply not only to the part layouts that are created when the project is first created, but to all future part layouts subsequently created in the project.
<i>Project files</i>	Under rare circumstances, Dorico could hang during project saving; this has now been fixed (macOS only).
<i>Project files</i>	When exporting flows from a project, Dorico no longer includes a thumbnail preview; previously, this preview could be for a flow or layout that was not included in the exported material, which could be confusing.
<i>Project files</i>	When saving a project as a template, Dorico now removes all MIDI CC and automation data in any preserved flows.
<i>Project files</i>	Under rare circumstances, an unpitched instrument in a percussion kit could be unexpectedly silent during playback; this has now been fixed.
<i>Script plug-ins</i>	The Lua interpreter on Windows requires filenames for loaded scripts to be encoded using the appropriate code page for the operating system's current region and language settings; Dorico now correctly encodes script filenames in the right code page, but this means that scripts with names that use characters outside the current code page will not load correctly. You are advised to stick to characters in the Latin-1 code page if possible (Windows only).
<i>Slurs</i>	When multi-pasting material including slurs to a different bar, Dorico no longer incorrectly links the newly pasted slurs at the new position to the original slur at its position.
<i>Staff labels</i>	Instruments that use player names as their staff label are no longer allowed to be grouped.
<i>Tempo</i>	When Dorico is set to show metronome marks to a specific level of precision, it could incorrectly display floating point values with leading zeroes: for example, <b>152.09</b> would display as <b>152.9</b> ; this has now been fixed.
<i>Tempo</i>	Under very rare circumstances, setting a sufficiently slow final tempo for a gradual tempo change could cause Dorico to crash; this has now been fixed.

Component	Issue
<i>Tempo</i>	Under certain circumstances, a gradual tempo that coincides with a time signature was no longer correctly aligned with the time signature in the presence of another tempo marking at its end; this has now been fixed.
<i>Tempo</i>	The abbreviation for <i>circa</i> in metronome marks has been changed to <i>ca</i> instead of <i>ca.</i> , i.e. the period (full stop) indicating the abbreviation has been removed.
<i>Ties</i>	Edits to the endpoints and curvature of a segment of a tie at the start of the system where the tied is between independent notes (for example, in different voices, or non-rhythmically contiguous notes) now correctly update the display immediately.
<i>Tonality systems</i>	Under certain circumstances, custom tonality systems in projects created in Dorico 3.5 or earlier could be overwritten with a different tonality system in the user library that happens to have the same internal identifier, causing unexpected effects when opening that project in the latest version; this has now been fixed.
<i>Tuplets</i>	Attempting to create a tuplet where either side of the ratio is zero no longer causes Dorico to crash.
<i>User interface</i>	The preview in <b>Library ▶ Paragraph Styles</b> now correctly displays foreground and background colors, if specified.
<i>User interface</i>	After saving a custom ensemble in the ensemble builder, if you switch to the <b>Choose</b> page without first closing the ensemble builder, the new custom ensemble now correctly appears there right away.
<i>User interface</i>	The ensemble builder popover now disappears when the application window is deactivated, such that it now behaves the same as the instrument popover.
<i>User interface</i>	The alignment of noteheads and flags to stems when using a standard resolution (i.e. non-Retina) display has been improved (macOS only).
<i>User interface</i>	When closing a project that was previously displayed full screen, an empty Space is no longer left behind (macOS only).
<i>User interface</i>	When opening a new window for an existing project, the Mixer button on the toolbar is now correctly enabled if appropriate.
<i>User interface</i>	If you specify a bar number that cannot be found in <b>Edit ▶ Go To</b> , Dorico will now navigate to the first system in the relevant flow, rather than doing nothing.
<i>User interface</i>	Key commands shown in the hands-on tutorials accessed via the <b>Learn</b> page of the Hub now show the appropriate shortcut modifier keys for macOS and Windows as appropriate.

Component	Issue
<i>User interface</i>	It is no longer possible to incorrectly specify negative durations for grace notes and tremolos on the <b>Timing</b> page of Playback Options.
<i>User interface</i>	The long-lamented disclosure arrow in the tree control in the <b>Key Commands</b> page of Preferences has returned home after a heart-wrenching absence of nearly 18 months.
<i>User interface</i>	If a popover is visible when the application loses focus, when the application regains focus, the popover itself regains focus, allowing you to continue typing into it without clicking in it again.
<i>User interface</i>	The bars in the system track would often not highlight when hovered over with the mouse; this has now been fixed.
<i>User interface</i>	When one of the right-hand panels is shown, it is now correctly updated to match the current selection state.
<i>User interface</i>	The single step size for the spin box paddles in the Zoom dialog is now 1%.
<i>User interface</i>	The <b>Follow selection changes on undo and redo</b> option on the <b>Note Input and Editing</b> page of Preferences now works as expected once more.
<i>User interface</i>	Longer instrument names are now elided in the track header in Play mode if necessary.
<i>User interface</i>	The layout selector in the toolbar is now correctly disabled both in Print mode and when editing page templates and flow headings in Engrave mode.
<i>Video</i>	Under some circumstances, Dorico could fail to connect the appropriate channel for the video soundtrack so that it can be heard and balanced via the Mixer when reopening an existing project; this has now been fixed.
<i>Video</i>	When playing back video during real-time recording, the video would jump forwards by 20 seconds compared to normal playback; this has now been fixed.





**Credit notice**

Steinberg, Dorico, WaveLab, Cubase, Nuendo, ASIO and VST are registered trademarks of Steinberg Media Technologies GmbH. VST Sound is a trademark of Steinberg Media Technologies GmbH. Windows is a registered trademark of Microsoft Corporation in the United States and other countries. Mac, Mac OS and OS X are trademarks of Apple Inc., registered in the U.S. and other countries. All other product and company names are trademarks and registered trademarks of their respective holders. All specifications are subject to change without notice. Made in Germany.

Copyright © 2023 Steinberg Media Technologies GmbH. All rights reserved.