

1. Loudness, Velocity and Volume.

In the 'real' world of music, whether an instrument is struck, bowed, twanged, picked, plucked, blown into or manipulated in any other way, the resulting loudness or softness is determined by the player. Depending on the instrument involved, in many instances a single note may have variations from very soft to very loud. Again, this is determined by the player.

In the 'virtual' world of computerised music, however, it is not so clear. A distinction must be made between different forms of computerised music. Wave files, as only one example from many, may be recordings of 'real' music. Here we are only concerned with music in MIDI format in general, and NoteWorthy Composer (NWC) in particular.

Loudness or softness is determined by a combination of Velocity, Volume and Expression. Each of these have a range of between 0 (zero) and 127.

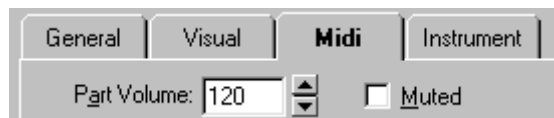
In MIDI music each note has only one velocity value (like a keyboard instrument) which occurs at the start of the note. It cannot be changed while the note is playing.

In NWC the use of velocity, as affected by dynamics or dynamic variations, is adequate for many notation purposes.

- Notating for keyboard or percussion
- Where no variation in loudness is required
- Variation in loudness or softness required over a series of short notes.

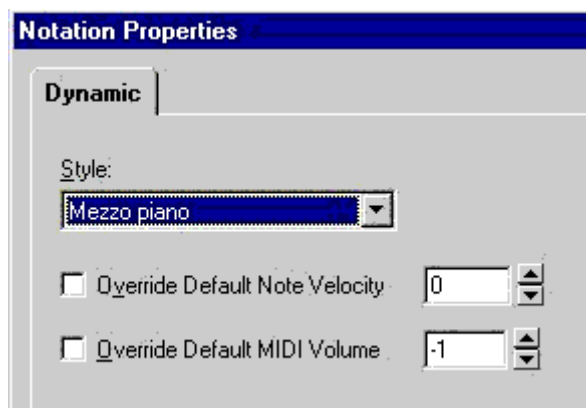
NWC uses dynamics, each with its own default velocity value, for ease of use. The default values used are: ppp = 10, pp = 30, p = 45, mp = 60, mf = 75, f = 92, ff = 108 and fff = 127.

The default volume can be edited in the staff properties window - thus:



It is my opinion that leaving the default at 127 in combination with the higher range of dynamic velocities can cause some distortion on some systems. Therefore I shall use 120 instead.

Both velocity and volume can be overridden at any position in the file by means of the Notation Properties window. Either insert a dynamic from the Insert menu or use the toolbar button then edit it from the Edit / Properties menu. The window that appears differs slightly in appearance depending on which of these methods you use.

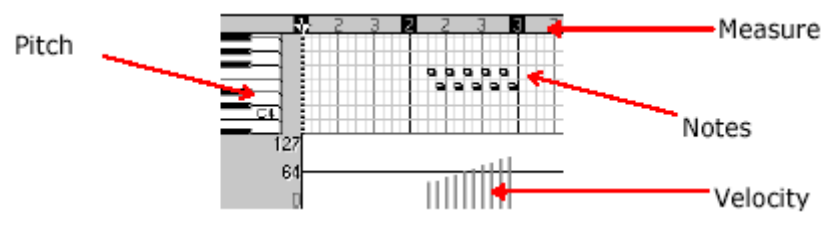


For ease of reference, I shall use the override function to display the default values in the editor.

The use of dynamics is the simplest way of making your music soft or loud. Using dynamic variations, such as crescendo, diminuendo or decrescendo to raise or lower the loudness of a series of notes is shown in this example:

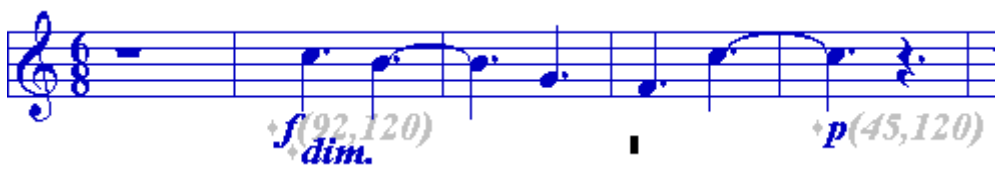


Here the use of dynamics and a dynamic variation is adequate. The phrase starts quietly and gradually gets louder. The effect can be seen graphically in MIDI sequencing software:

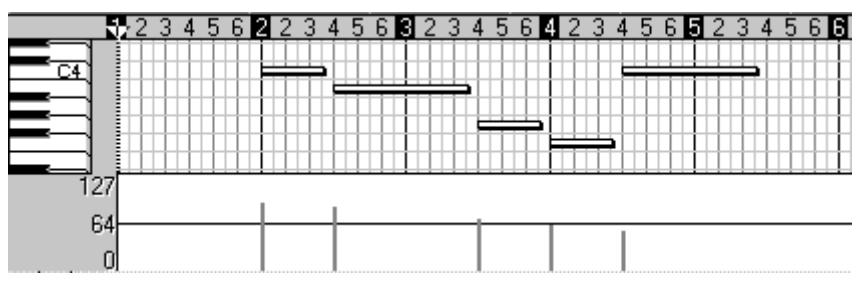


Notice the smooth transition from 45 (p) to 92 (f) represented by the vertical Velocity lines.

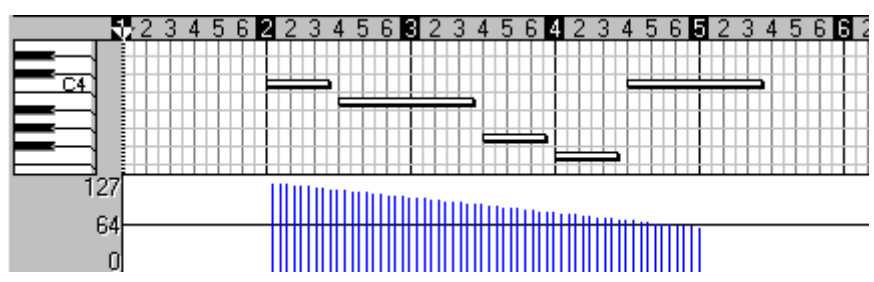
With notes of greater duration, as in the following example, the transition is less smooth:



The second note (B) retains its velocity throughout its duration and the G is considerably quieter, so instead of a smooth transition there is a sudden drop in loudness.



What we really need to achieve with this phrase is:



This effect is produced by the use of the Volume MIDI Controller. Each of the blue vertical lines represents an instruction sent to the soundcard, or other playback device, to change the output volume.

In NWC there are three different methods which produce exactly the same result.

- Overriding the default dynamic properties, both velocity and volume, in combination with dynamic variations
- Using Volume MIDI Controls
- Using Expression MIDI Controls.

In all of these cases the same values are used and it is these values which must be established before proceeding.

Establishing values.

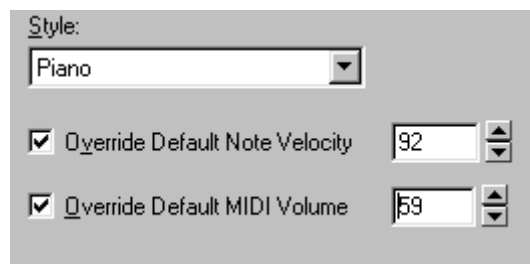
As has been shown, each dynamic has two values - velocity and volume - shown as bracketed figures in the editor - dynamic(velocity,volume), but not visible in print-outs, NoteWorthy Player or browser plug-ins. If we need to use MIDI controllers to raise or lower the volume of sustained notes we must know the relative values involved.

Let us start by determining the relationship between forte and piano. We know that in NWC, forte has a default velocity of 92 and that piano has a default velocity of 45.

Create a new NWC file with one staff. Select a woodwind instrument, such as an oboe, enter 3 half notes of the same pitch, and a starting dynamic of p:



Play the file. Turn your speaker volume to a comfortable level so that you can hear the note clearly but not too loudly. Now we want to find out the volume that will enable a dynamic of f to emulate a p - for reasons which will become clear later. Insert a second dynamic at the second note, and edit it thus:

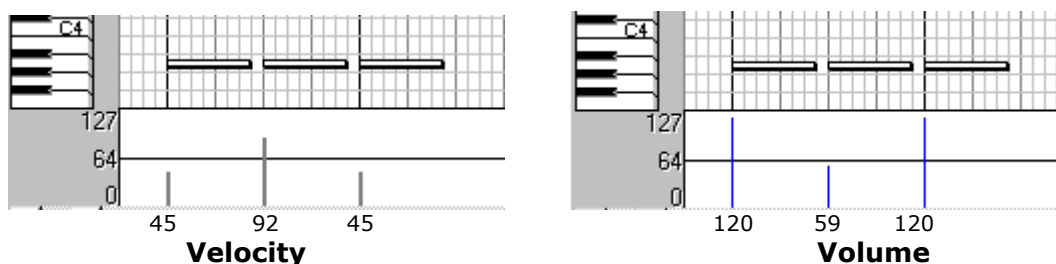


Finally, insert a dynamic at the third note, set back to the default values of p. Your staff should now look like this: (N.B. I have varied the vertical positions for visual clarity)



and when the file is played, all three notes should sound as if they were 'p'.

Let us take a look at what is actually happening.

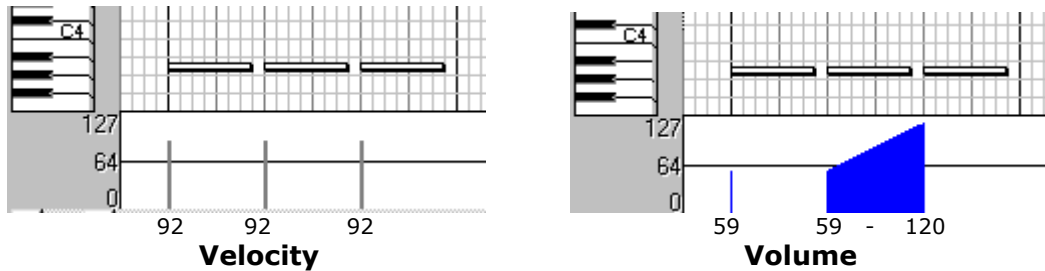


Now edit the first dynamic; replace the centre dynamic with a dynamic variation (crescendo); and change the third dynamic thus:



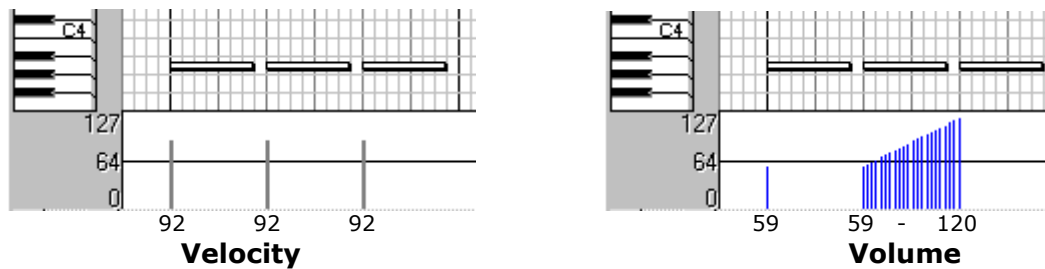
Play and listen.

Look at what is going on now:



Notice how closely packed the volume commands are. There would be less drain on your resources if, instead of using dynamics alone to control volume, you were to use the volume mpc.

We shall be working on the mpc window a little later, but to demonstrate the difference, here is the visual result that can be obtained without affecting the sound:



Over a short distance like this, therefore, an mpc has an advantage in that it can be more sparing of resources.

It would be nice if all music was composed using only dynamics of f(92) and p(45), with all variations being between those two. In reality, however, this is not the case.

In every phrase or section of music where a Volume MIDI control is used, the velocity of all the lower dynamic(s) must be overridden to that of the highest. The volume must be returned to its base value before dynamics alone can be used to control the loudness of the music.

As has been demonstrated, it is the relationship between Velocity and Volume which determines how loud or soft a series of notes sound. This can be calculated mathematically, using the following formula:

$$\frac{\text{Lower default velocity} \times \text{Default Volume}}{\text{Higher default velocity}} = \text{Required Volume}$$

Using p and f, this would be:

$$\frac{45 \times 120}{92} = 58.7 \text{ (rounded up to 59)}$$

For ease of reference I produce below a conversion chart based on a part volume of 120 and the default NWC dynamics. For any section of music where a dynamic variation is required, select the highest dynamic to be used from the top line and read down to

find the volume which will reduce it to the equivalent of the dynamic in the left hand column.

| | fff | ff | f | mf | mp | p | pp |
|-----|-----|-----|-----|-----|-----|-----|-----|
| pp | 28 | 33 | 39 | 48 | 60 | 80 | 120 |
| p | 43 | 50 | 59 | 72 | 90 | 120 | |
| mp | 57 | 67 | 78 | 96 | 120 | | |
| mf | 71 | 83 | 98 | 120 | | | |
| f | 87 | 102 | 120 | | | | |
| ff | 102 | 120 | | | | | |
| fff | 120 | | | | | | |

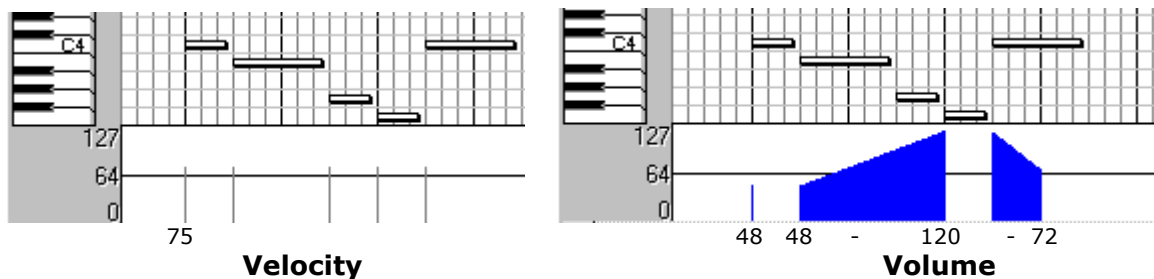
Once you have grasped the basics, you may prefer to use alternative dynamic / volume variations.

In NWC it is possible to insert a single dynamic at the beginning of the staff and control the loud / soft variations of the entire work using only MIDI controllers. If you should decide to do this, you should ensure that dynamics, either overridden or as text, are included in the work for the benefit of others.

Just as an example of how these values can be used here are two more examples, using the same musical phrase as before:



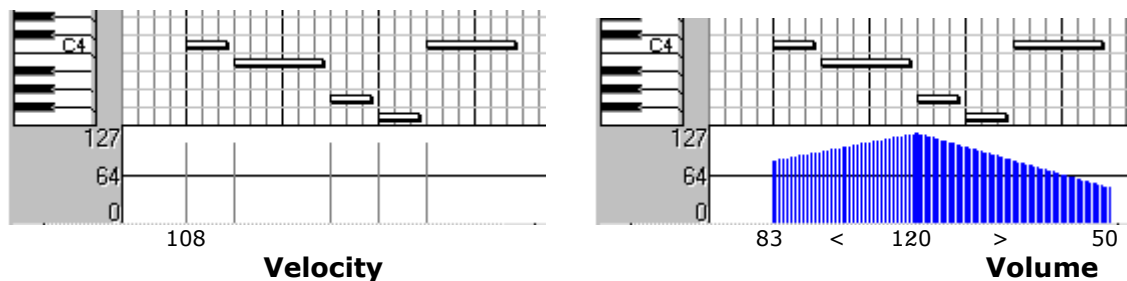
As can be seen, a velocity of 75 (default for mf) is used for both pp and p whilst only the volume is adjusted.



Or again:



The velocity here, again, matches that of the loudest dynamic.

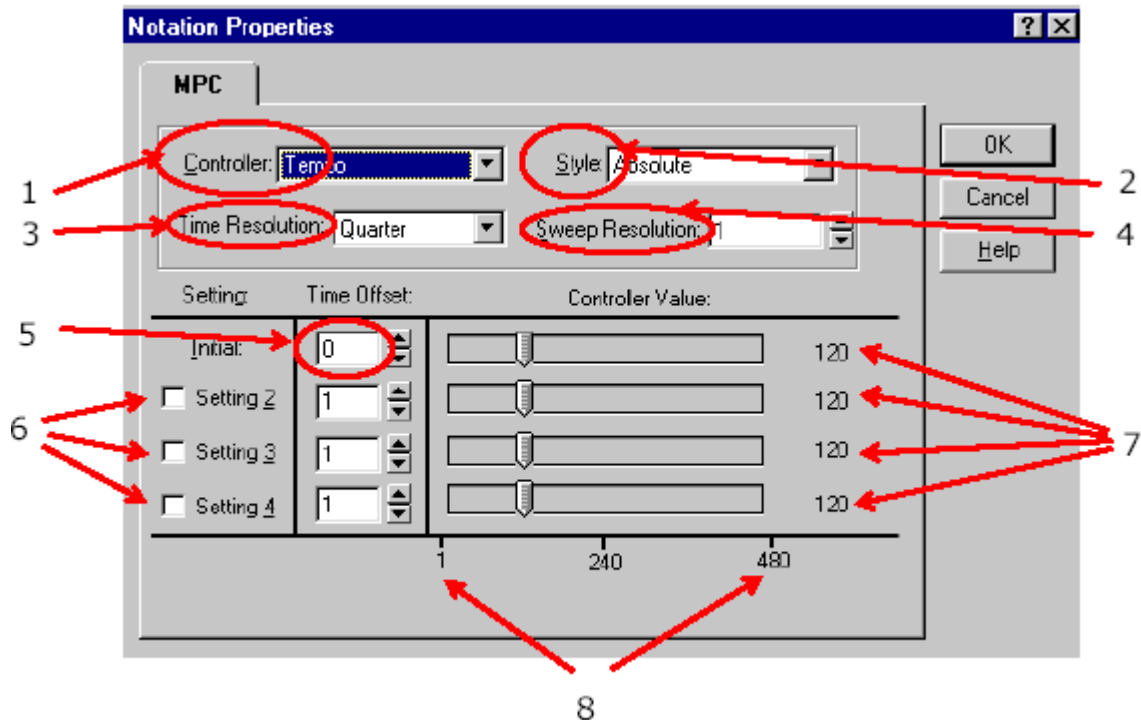


There are many more similar variants encountered in musical scores and, all other things being equal, in most instances the use of dynamics and their variations in combination with volume is adequate. There are, however, occasions when this is not sufficient. This is where using the NWC MPC dialogue window has the advantage.

The next section of this guide, is designed to familiarise you with this sometimes daunting facility.

2. Multi-point Controllers (MPC).

At first glance, the mpc dialogue window can be a bit daunting to the newcomer, but by breaking it down to its various components the logic behind it is very simple to understand.



1. Controller.

There are several MIDI controllers to choose from but, with minor differences, all are identical in the way they are used. We shall be looking at only a few of them here.

2. Style.

There are only two styles, Absolute and Linear. Absolute is used to send single commands, one at a time, to the playback device. Linear, on the other hand, sends a whole series of commands for a period of time specified by the value in the Settings boxes (6).

3. Time resolution.

The options here are all related to note duration, and whatever is selected here becomes the unit by which everything else is measured.

4. Sweep Resolution.

There may be times when a large linear change is required over a short distance. In these cases far more commands are generated than is either necessary or desirable.

5. Initial Setting.

This setting is, in effect, a delay function. It may not always be possible to insert a controller at the precise point where a change is to occur - such as halfway through a note. It counts as the first point. If it has no value (zero) the control takes effect, or starts, at the insertion point.

6. Settings boxes

These are subsequent points at which a change starts or finishes.

7. Controller Values

The figures shown on the right end of the sliders change as the sliders move, allowing for both flexibility and accuracy. After moving the slider as near to the desired value as possible using the mouse, it can then be 'tweaked' using the left or right arrow keys.

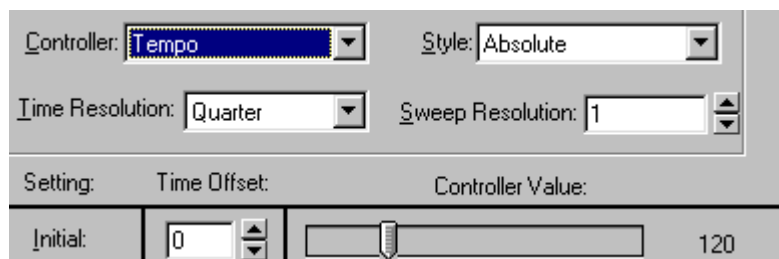
8. Controller Range.

The figures here may vary from controller to controller, signifying the range over which it can be manipulated.

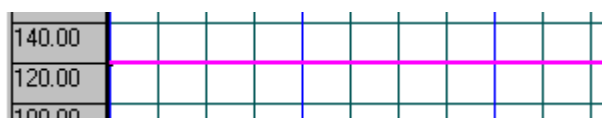
In order to familiarise the beginner with the various options of the mpc window, we will start with what must be the easiest Controller of all:

Tempo.

The Tempo mpc operates in exactly the same way as other Controllers.



By inserting an unchanged default mpc command, the effect is exactly the same as inserting ♩=120 from either the toolbar button, or Insert / Tempo.



As shown here, the actual tempo control is displayed as a straight line, emanating from a single point.

Although tempo and tempo variances can be easily inserted into NWC without the need to use the mpc., there may be occasions when doing so produces a tidier score. As with all MIDI controllers, selecting 'absolute' in the dialogue box defines a continuous value until a further command is encountered.

In the following example, a pause is signified by a fermata:

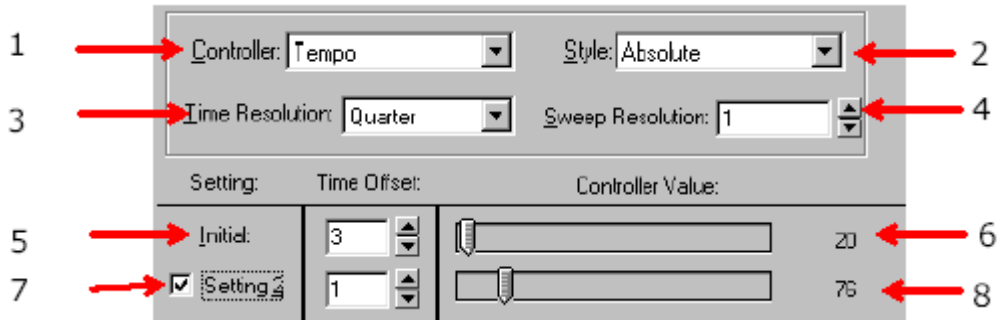


Fermatas have no effect when the nwc work is saved as a MIDI file, so the tempo needs to be adjusted to take the pause into account. The above image is taken from an orchestral score, where only the Basses and Celli are playing. All other staves have the fermata centred on the whole rest - as in the upper staff shown here.

This can be handled in one of two possible ways: 1) by inserting a new tempo, using rests to define the exact point of the change, then re-establishing the original tempo:

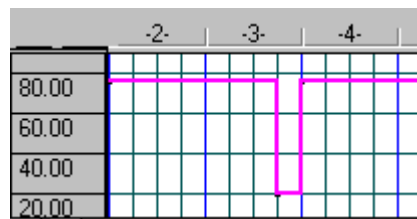


or by the insertion of a tempo mpc at the beginning of the bar in which the change is to occur, using the fermata for visual effect:



- 1. Select Tempo
- 2. The style is Absolute - no gradual transition required.
- 3. The Time Resolution is Quarter - beats per minutes is defined in quarter notes.
- 4. Sweep resolution 1 - not applicable, not changed
- 5. Initial - 3 Quarter notes delay before any change to take place
- 6. Slider 1 - value (bpm) of 20 to occur after the delay
- 7. Setting 2 - 1 Quarter note during which the 20 bpm is in effect
- 9. Slider 2 - the original tempo of 76 bpm restored.

Whichever of the two methods is employed, the effect is identical:



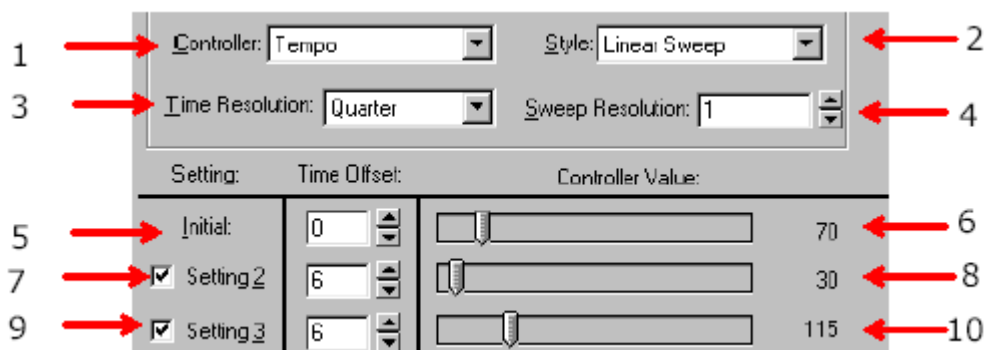
The only difference is the visual effect in the score.



In the following example, the tempo is set at 70, slows down between measures 2 and 4 to 30, then immediately increases to become 115 at measure 6. (All values relate to quarter notes - or crochets)

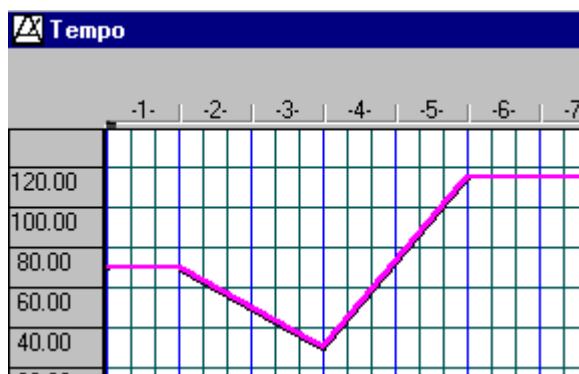


Alternatively, you could replace all but the starting tempo with a single tempo mpc at measure 2:



- 1. Select Controller
- 2. Select Linear sweep
- 3. Time resolution is still in quarter notes
- 4. Leave this at the default.
- 5. No delay - First sweep commences at insertion point
- 6. Starting value - as previously defined
- 7. This sets the duration (in quarter notes) of the Linear sweep between slider one (6) and slider 2 (8)
- 8. The value to be attained at the end of the first sweep
- 9. As 7, but relates to sliders 2 (8) and 3 (10)
- 10 The final goal of 115 is reached.

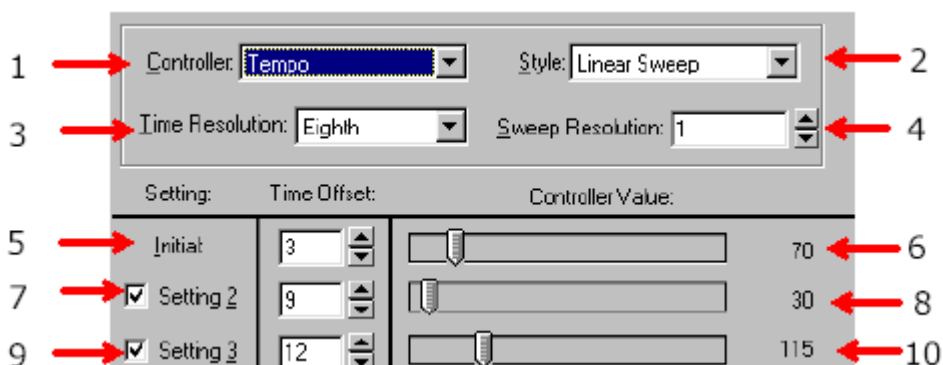
Result:



You may want the 'ritard' to start halfway through a measure - which in this instance would be the centre of the second beat. You could, if you wanted, split the quarter into two tied eighths but that would spoil the appearance.



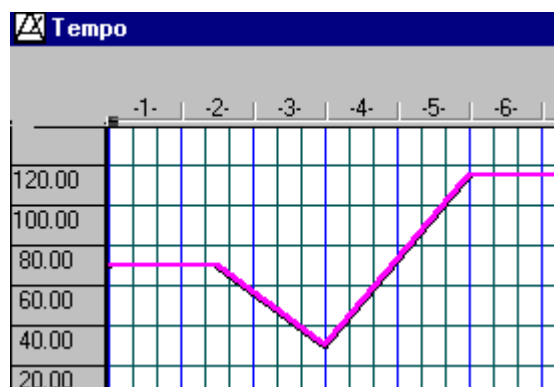
An alternative would be to insert an mpc at the beginning of measure 2:



There are several differences to be made between this set of controls and the preceding one.

- 1. Select Controller
- 2. Select Linear sweep
- 3. Time resolution is now reduced to eighth notes. (See 5, below)
- 4. Leave this at the default.
- 5. Because the onset of the control does not occur at the insertion point it needs to be delayed. The actual note duration of the delay is one and a half quarters - not possible as only whole numbers can be used. Therefore the 'Time resolution' needs to be changed to 'Eighth' (quaver), and everything else doubled - there being 2 quavers to a crochet..
- 6. Starting value - as previously defined
- 7. This sets the duration (in eighth notes) of the Linear sweep between slider one (6) and slider 2 (8)
- 8. The value to be attained at the end of the first sweep
- 9. As 7, but relates to sliders 2 (8) and 3 (10)
- 10 The final goal of 115 is reached.

Result :



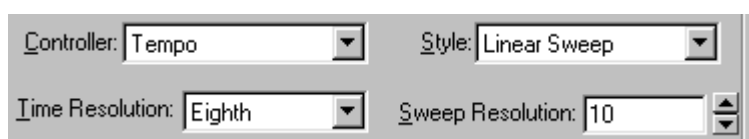
As you can see, the downward slope now starts in the middle of measure 2.

Again, the only difference is in the appearance of the score.



The final settings box and slider of the mpc window works in exactly the same way as the previous two, so the only option left to be explored is the 'Sweep resolution'.

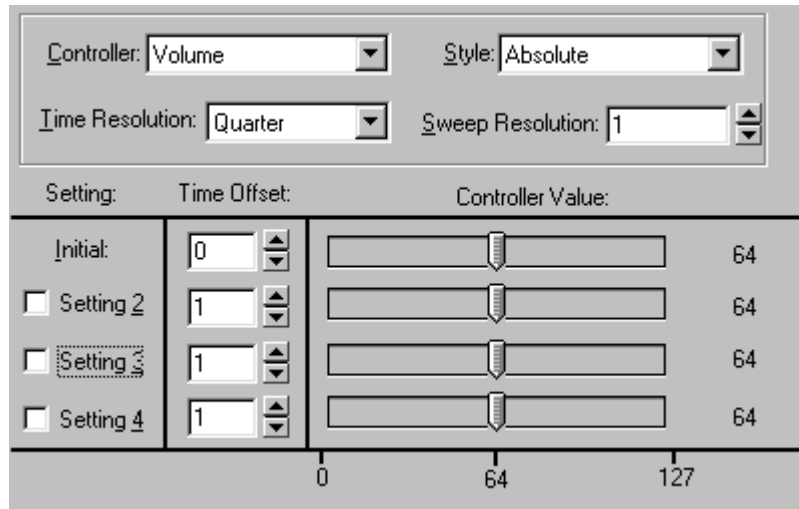
Leaving all others setting unchanged, a 'Sweep resolution' of 10.....



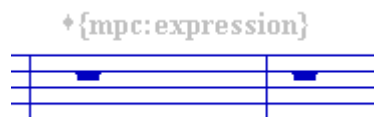
.... looks like this :

Volume & Expression Controllers

Volume and Expression MIDI controllers work in identical ways and are interchangeable. The Multi-point Controllers for these are almost the same as Tempo. The only difference being the range - 0 to 127.



Both Controllers have a starting default of 127. The Volume value can be changed in the Staff properties dialogue window, but the default Expression value can only be changed within the editor.



If Expression is used to control the loudness of the music, and if the default value of 127 is not going to be used, then a new default value must be inserted on all staves - or all used channels if there are more than one staff per MIDI channel. If more than one staff is on a single MIDI channel, then any changes on one staff affects the other(s).

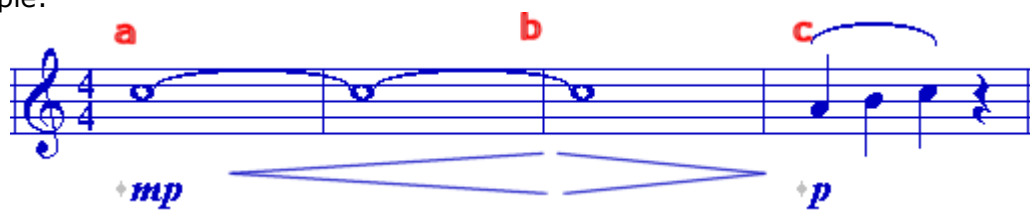
We have already seen how volume can be used in NWC by the use of dynamics. Now we will use one of those examples to demonstrate the use of Expression MIDI controllers.



In this case, only the velocity of p needs to be overridden to match that of f. The 'cresc' is for appearance only and has no effect because the velocity on either side is the same. The insertion **1** is as above, where the absolute value is 120.

The 'Minimum value required' box will then give the value required to reduce the higher dynamic to the lower.

Example:



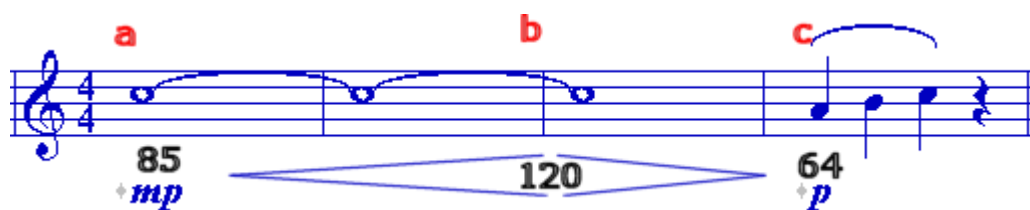
Here you will need three values: a, b and c. The required loudness of b may not correspond exactly to any default dynamic, but fall somewhere between *mf* and *f* - say 85. Enter 85 in the *Velocity* box for the *Highest Dynamic*; check that the *Maximum Value Required* matches either the staff volume or the *Expression* value inserted at the start of the staff (if none has been inserted the default is 127) and then select *mp* in the *Lowest Dynamic* box.

| | | |
|-----------------|----------|------------------------|
| Lowest Dynamic | Velocity | Maximum value required |
| <i>mp</i> | 60 | 120 |
| Highest Dynamic | Velocity | Minimum value required |
| <i>f</i> | 85 | 85 |

Then repeat for *p*:

| | | |
|-----------------|----------|------------------------|
| Lowest Dynamic | Velocity | Maximum value required |
| <i>p</i> | 45 | 120 |
| Highest Dynamic | Velocity | Minimum value required |
| <i>f</i> | 85 | 64 |

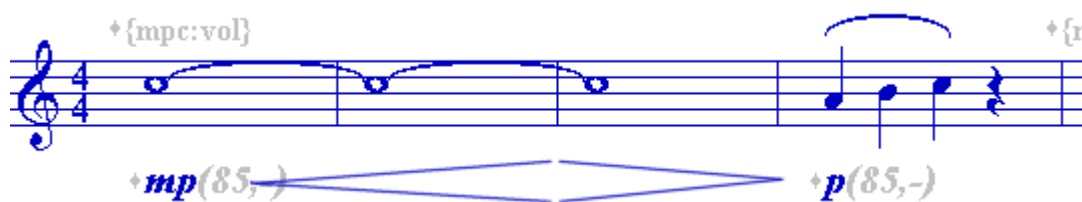
The values, taken from the *Minimum Value Required* box, are 85 for *mp*, 64 for *p*.



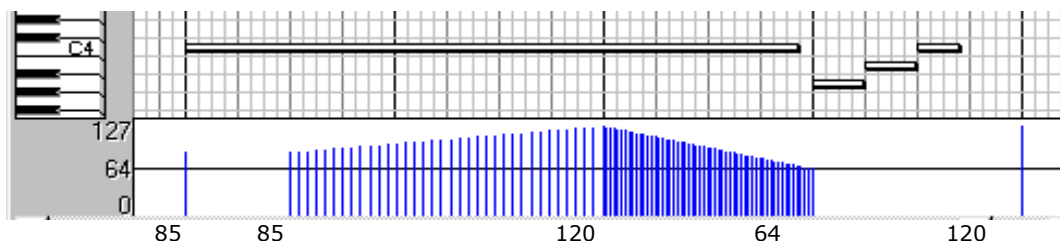
Override the velocities of both *mp* and *p* to 85 - being the highest velocity required and you are ready to insert MIDI controllers using the MPC dialogue window.

| | | | |
|---|--------------|-------------------|--------------|
| Controller: | Volume | Style: | Linear Sweep |
| Time Resolution: | Quarter | Sweep Resolution: | 1 |
| Setting: | Time Offset: | Controller Value: | |
| Initial: | 0 | | 85 |
| <input checked="" type="checkbox"/> Setting 2 | 2 | | 85 |
| <input checked="" type="checkbox"/> Setting 3 | 6 | | 120 |
| <input checked="" type="checkbox"/> Setting 4 | 4 | | 64 |
| | | 0 | 64 |
| | | | 127 |

As the values required for both Volume and Expression are identical, either may be used. The results would be the same.



Do not forget to reset the default value at the end of the section. This will enable you to use 'normal' dynamic velocities thereafter.



You may be wondering why the Initial value was not used to delay the commencement of the sweep, rather than have two points at the same value. In this particular instance, the change to 85 was required at the insertion point. Had the Initial value been set at 2 the first two quarters would have sounded at the default volume (120) followed by a sudden drop to 85 halfway through the whole note. Had there been a previous value of 85 then the delay function could have been used.

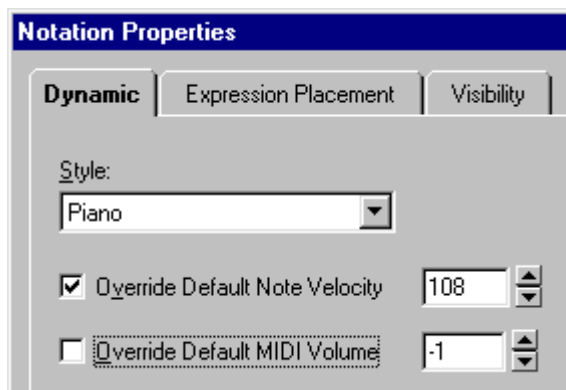
In this example, we want the phrase to start at *p*; remain constant for 3 half notes, then increase in volume to *ff* at the beginning of measure 9.



We already know - from the section on velocity - that using velocity alone will not produce the desired effect. In this instance the loudest part of the phrase is *ff* (108). Replace the *p* with *ff*...



... but now it looks wrong - so edit the *ff* you just inserted so that it 'looks' like a *p*. The easiest way to do this is to first tick the Override Note Velocity box (retaining the value 108) then change the Style to Piano:



Now the score looks like this:



A visual improvement, but not yet functional. First, by using Volcalc, determine what volume turns an ff into a p.

| | | |
|-----------------|----------|------------------------|
| Lowest Dynamic | Velocity | Maximum value required |
| p | 45 | 120 |
| Highest Dynamic | Velocity | Minimum value required |
| ff | 108 | 50 |

With your cursor to the left of the p, insert a Multi-Point controller.

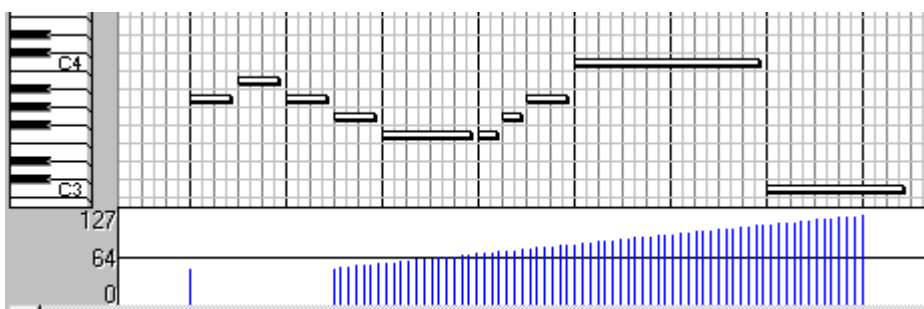
| | | | |
|---|--------------|-------------------|--------------|
| Controller: | Volume | Style: | Linear Sweep |
| Time Resolution: | Quarter | Sweep Resolution: | 1 |
| Setting: | Time Offset: | Controller Value: | |
| Initial: | 0 | 50 | |
| <input checked="" type="checkbox"/> Setting 2 | 6 | 50 | |
| <input checked="" type="checkbox"/> Setting 3 | 22 | 120 | |
| <input type="checkbox"/> Setting 4 | 1 | 64 | |
| | | 0 | 64 |
| | | | 127 |

The figure in Setting 2 box is calculated by the number of quarter notes over which the value of 50 remains constant. Setting 3 is the number of quarter notes over which the sweep occurs. The simplest way to do this is to count the number of whole bars affected, multiply by the number of beats (quarters) in a bar and add any extras at either end. (5 bars x 4 quarter notes + 2 quarter notes in bar 2 = 22)

The score should now look like this:



The MIDI commands sent to the sound device during playback are represented thus:



If (as is to be hoped) you have carried out the experiment in NWC, you will also hear the result.

