

Vectorstylers Leading

From a traditionally pre-digital point of view to Pagemaker, CorelDraw, QuarkXPress and MS Word, leading is a paragraph attribute. If we look at Vectorstyler's biggest competitors there is a difference in approach and division when you look at different software packages.

Adobe treats leading as a character attribute, which means that you can apply more than one leading value within the same paragraph. The largest leading value in a line of type determines the leading for that line. (However, you can select a preferences option so that leading applies to the entire paragraph, instead of to text within a paragraph. This setting does not affect the leading in existing frames.)

Affinity treats leading as a paragraph attribute and a 'Leading Override' in the Character panel (does not affect the paragraph but only the lines it is applied to.) Affinity is most similar to Vectorstyler and mixes up the paragraph property with a questionable 'leading character override' that no one understands or uses.

I asked the developer whether the choice to make leading a character attribute is a conscious choice and what his reasoning is for it:

"When the VS leading attribute was designed, this disparity between applications was considered. That is why there is leading both as a character and as a paragraph attribute."

Also, the paragraph level leading was designed to be relative only (can be changed of course), because the actual leading info comes from the font height by default, which is a character attribute, and there can be multiple character styles inside a single paragraph, so it was better to refer to these in terms of percentage."

As far as I know, relative/auto leading either takes the information from the first character of a line or the largest character and applies it to the entire line. And because leading uses the information from character metrics height as a reference, makes leading not automatically a character attribute.

What do we use relative/auto leading for?

Virtually all word-processing and page-layout programs have some leading or line spacing specification that they use by default. If you don't specify a particular leading value, the program will use its default value. When you first open a page layout program, this default line spacing is set to automatic leading.

By using auto leading, you can change your text size as many times as you like and the leading will adjust proportionally and automatically. This is a real time-saver when you're unsure of your final point size and want the freedom to experiment.

As soon as you want full control over your typography, leave auto/relative) leading and switch to an exact definition. InDesign does this automatically. When you change the Auto leading (value between brackets) you automatically switch to exact.

Double specification

In its penchant for excess, Vectorstyler also seems to opt for a mix up. A choice that results in an incomprehensible mix of leading parameters. Looks like compiled by someone who has neither received typographical training nor has ever made page layouts himself.

The major problem with defining leading in two attributes is that they do come together in one appearance and they also influence each other. Paragraph leading influences character leading and vice versa.

With the thought in mind that the developer stated earlier in the forum: "One thing to keep in mind is that VS is a drawing app and typography might be used in a non-traditional way." it is good not to make the leading more complicated than necessary. Vectorstyler is not intended for newspaper design. Leaving conventions is also not a problem as long as it works well and is understandable.

Whether it belongs to paragraph or character doesn't really matter to me. Leading as an a character attribute (Adobe approach) seems to be the most flexible. Specify it at the smallest/lowest level, the characters. Although theoretically incorrect, it is also better for user convenience to place leading under 'character'. After all, you also say 10/12 or 10/14. You don't want to open different panels for that.

I think adjustment and simplification is appropriate. Hoping for the best i'm trying to come up with a few thoughts..

1. The current situation and it's problem

1

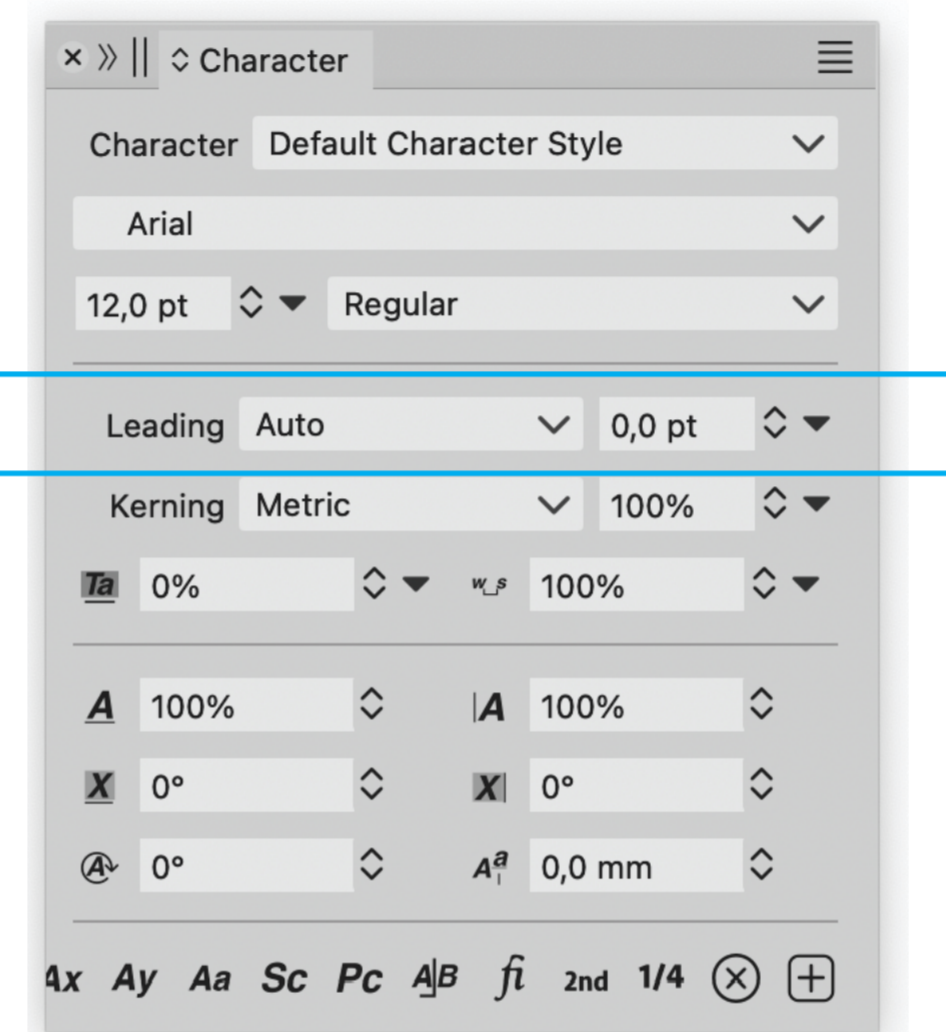
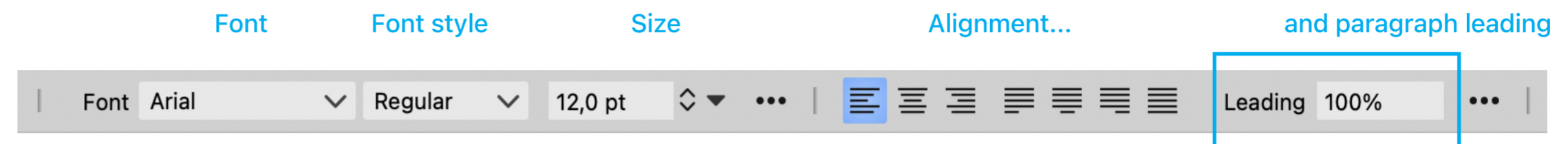
This context bar (as it is now) line up is correct for me. To setup new text, this is what you need...

2

Setting up text from the Character and Paragraph panels is different. The design is confusing, misleading and encourages incorrect use of leading due to incorrect hierarchy and placement in the Character and Paragraph panel.

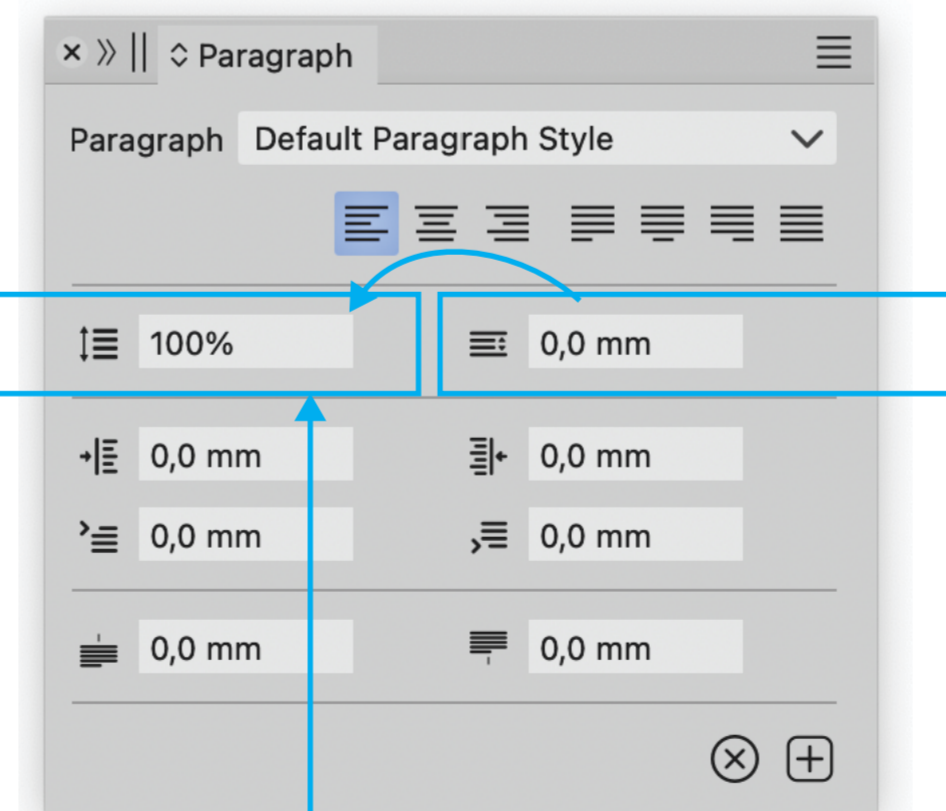
3

The problem is the visual prominence or hierarchy and setting options are opposite to how it is used.



Secondary (Character) leading
Sets a leading justification value at character, word(s) and sentence level

Visual appearance
Prominent placement
Extensive
Named



Primary (Paragraph) leading
Sets the base leading on paragraph level

Visual appearance
Subordinate placement
Minimal
Unnamed

Extra Line Spacing in addition to the leading.
I don't understand what this is for. Very strange. Cannot find any use case example for this.

When you see this for the first time, you assume that this is a relative value. A kind of Auto value.

This setting is the same as the setting in the paragraph panel.

The Paragraph Style options panel has an option to set to 'Fixed' instead of Auto.

2. Alternative design for Leading as an Paragraph and Character attribute

While I do not see the need for Vectorstylers 'Dual Leading System' in both Paragraph and Character (cannot find any use case example), I propose to stay as close as possible to the current design. I envision the following.

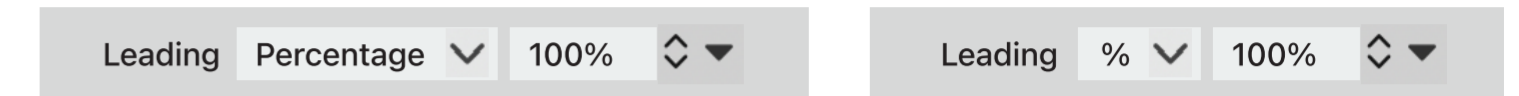
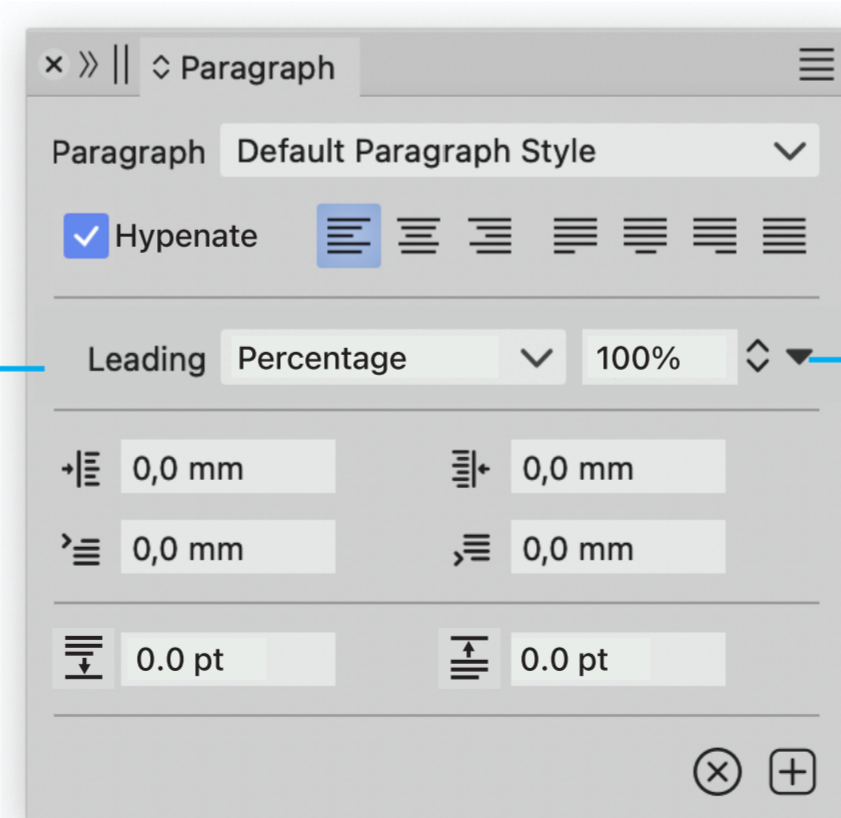
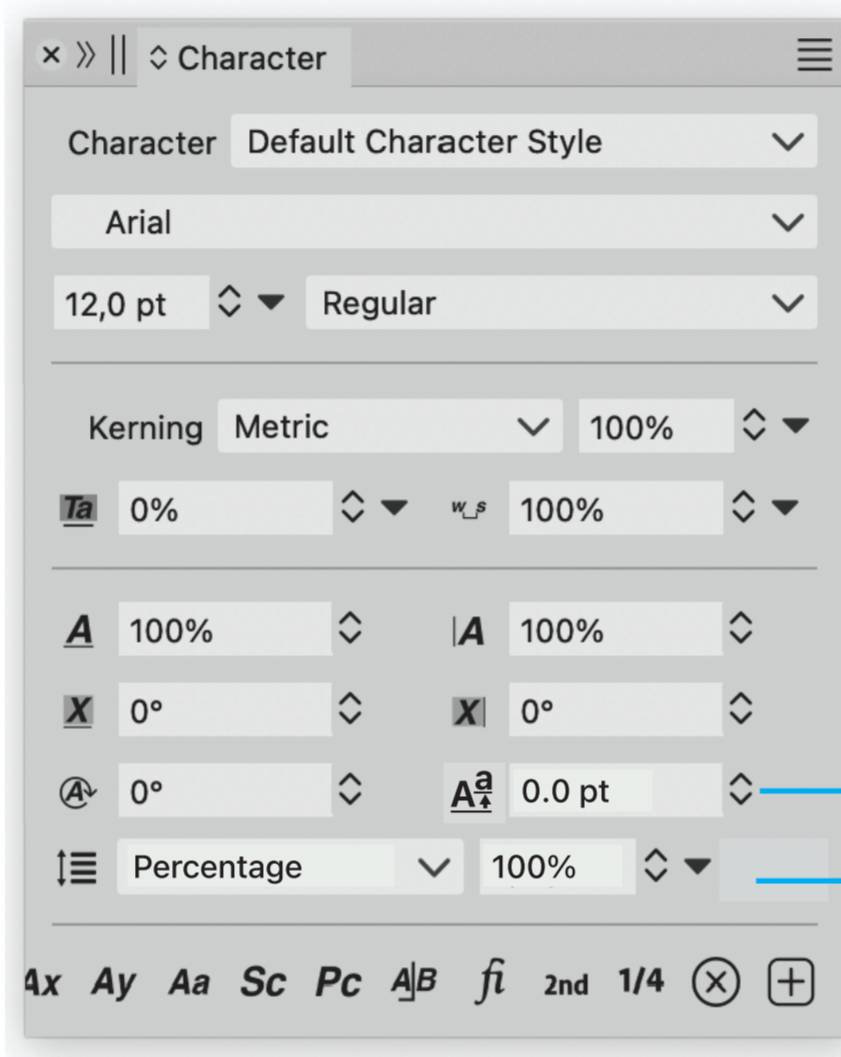
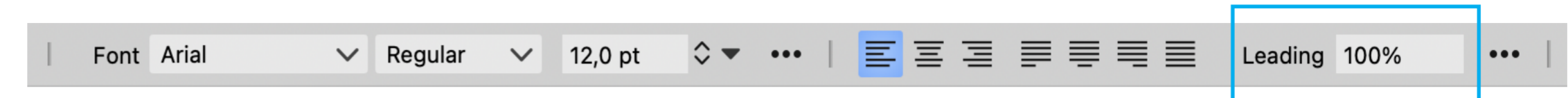
Give visual prominence to the primary leading setting and make the secondary leading appear secondary. A simple solution by exchanging the visual appearance. Form follows function.

Although you see little difference at first glance, the visual hierarchy of the leading in both panels is now reversed and both extended with leading method selector.

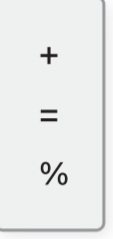
Secondary leading
Sets a leading justification value at character, word(s) and sentence level just like Baseline Shift.

(And possible for me to ignore in this subordinate position.)

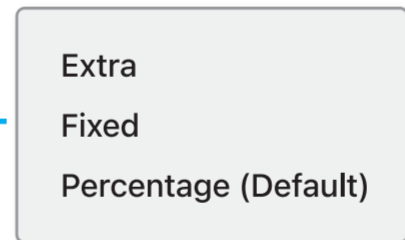
Primary leading
Sets the base leading on paragraph level



The question is whether leading selector should also be visible in the context bar or simply follow the setting in the panel with the corresponding value selector. If so, two options for Leading in the context bar



Baseline Shift



Similar Leading methods
Selection for paragraph as for the character leading mode: percentage (relative), Fixed (absolute pt), or Extra (added pt).

Percentage	Fixed (pt)	Extra (pt)
0%	4.0	-10.0
10%	6.0	-9.0
20%	8.0	-8.0
30%	8.0	-7.0
40%	9.0	-6.0
50%	10.0	-5.0
60%	11.0	-4.0
70%	✓ 12.0	-3.0
80%	14.0	-2.0
90%	16.0	-1.0
✓ 100%	20.0	✓ 0.0
110%	22.0	+1.0
120%	24.0	+2.0
130%	36.0	+3.0
140%	48.0	+4.0
150%	60.0	+5.0
160%	72.0	+6.0
170%	96.0	+7.0
180%	144.0	8+.0
190%	288.0	9+.0
200%		+10.0

Relative and Extra based on font metrics ascender and descender, not the so-called 'default line-height' which is basically a font designer's personal preference which is irrelevant. Leading is highly dependent on line width and application.

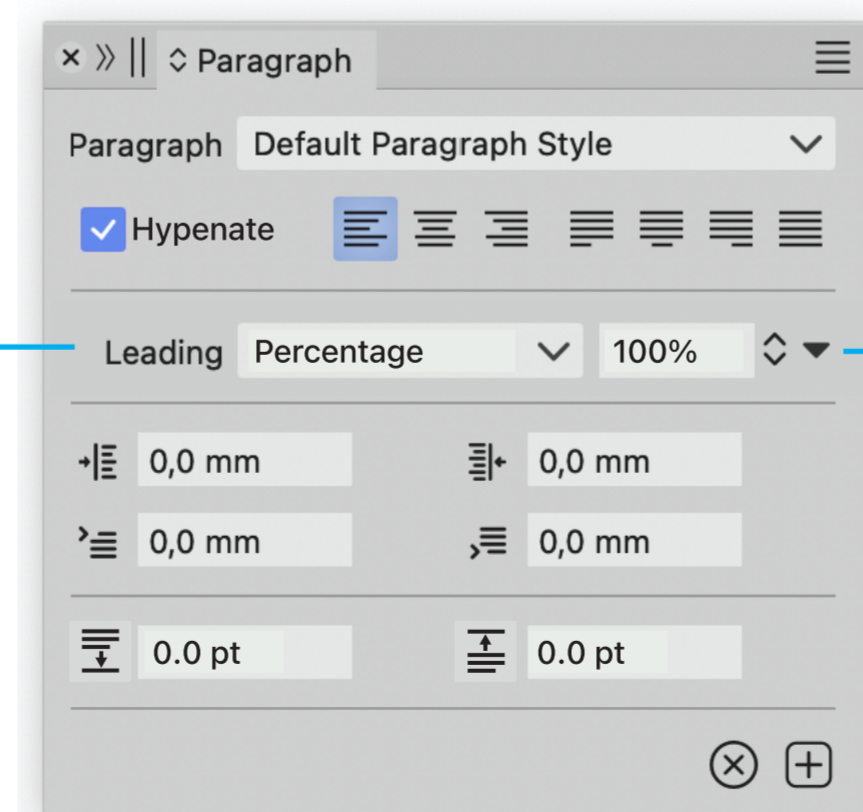
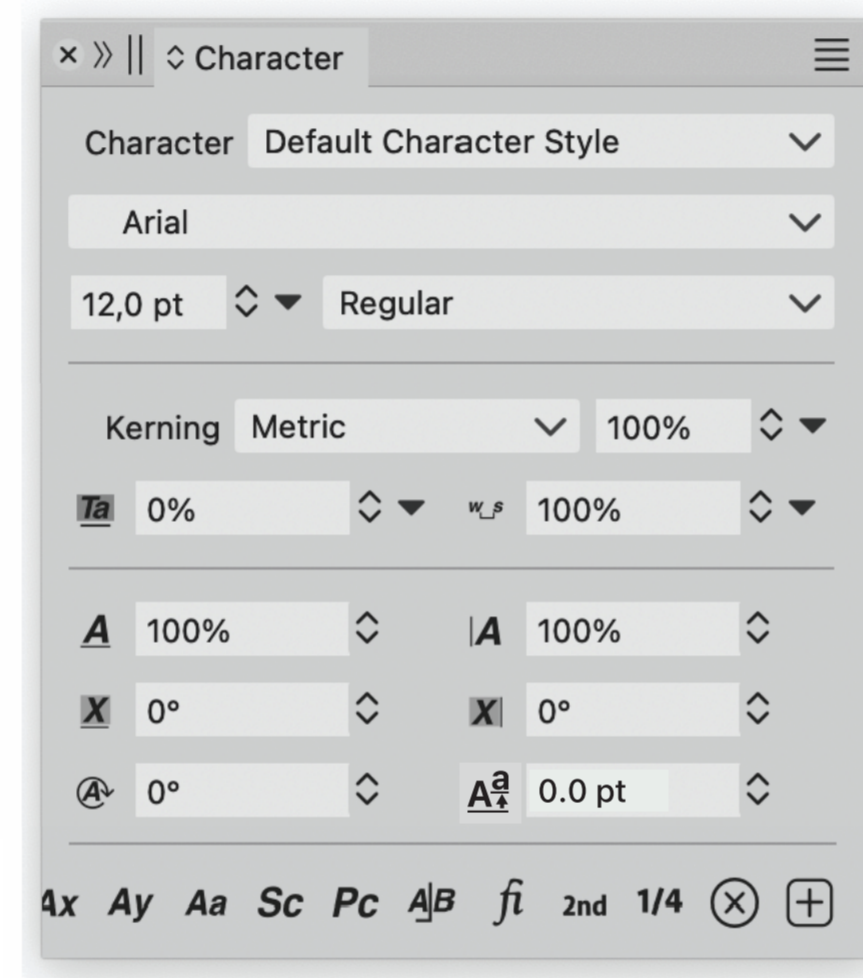
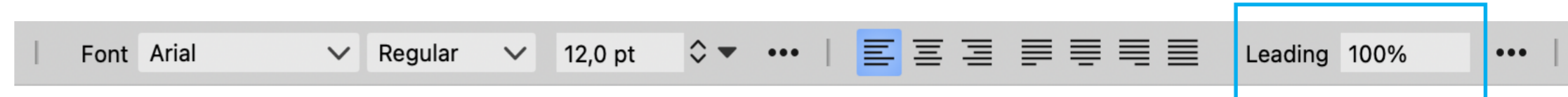
Note!

- Baseline Shift and Space Before and After Paragraphs have new icons and units are set to Points! (see next chapter Units)
- I also found a nice place for a hyphenations checkbox.

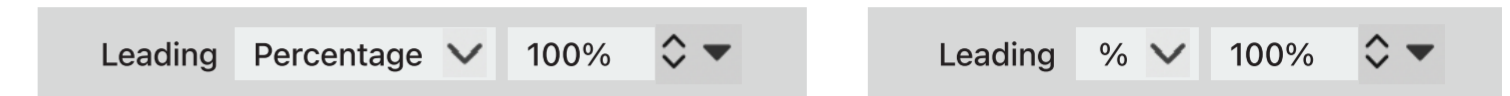
3. Leading as an Paragraph attribute

Solid more classic approach where leading is housed at Paragraph (like QuarkXPress).

The disadvantage of leading as a paragraph attribute is that you can apply no more than one leading value within the same paragraph.



Leading
Sets the leading on paragraph level



- Extra
- Fixed
- Percentage (Default)

The question is whether leading selector should also be visible in the context bar or simply follow the setting in the panel with the corresponding value selector. If so, two options for Leading in the context bar

- +
- =
- %

Leading methods
Selection for leading mode:
percentage (relative), Fixed (absolute pt), or Extra (added pt).

- Extra
- Fixed
- Percentage (Default)

Note!
- The default percentage leading should have a setting at app or document level.

Percentage	Fixed (pt)	Extra (pt)
0%	4.0	-10.0
10%	6.0	-9.0
20%	8.0	-8.0
30%	8.0	-7.0
40%	9.0	-6.0
50%	10.0	-5.0
60%	11.0	-4.0
70%	✓ 12.0	-3.0
80%	14.0	-2.0
90%	16.0	-1.0
✓ 100%	20.0	✓ 0.0
110%	22.0	+1.0
120%	24.0	+2.0
130%	36.0	+3.0
140%	48.0	+4.0
150%	60.0	+5.0
160%	72.0	+6.0
170%	96.0	+7.0
180%	144.0	8+.0
190%	288.0	9+.0
200%		+10.0

Relative and Extra based on font metrics ascender and descender, not the so-called 'default line-height' which is basically a font designer's personal preference which is irrelevant. Leading is highly dependent on line width and application.

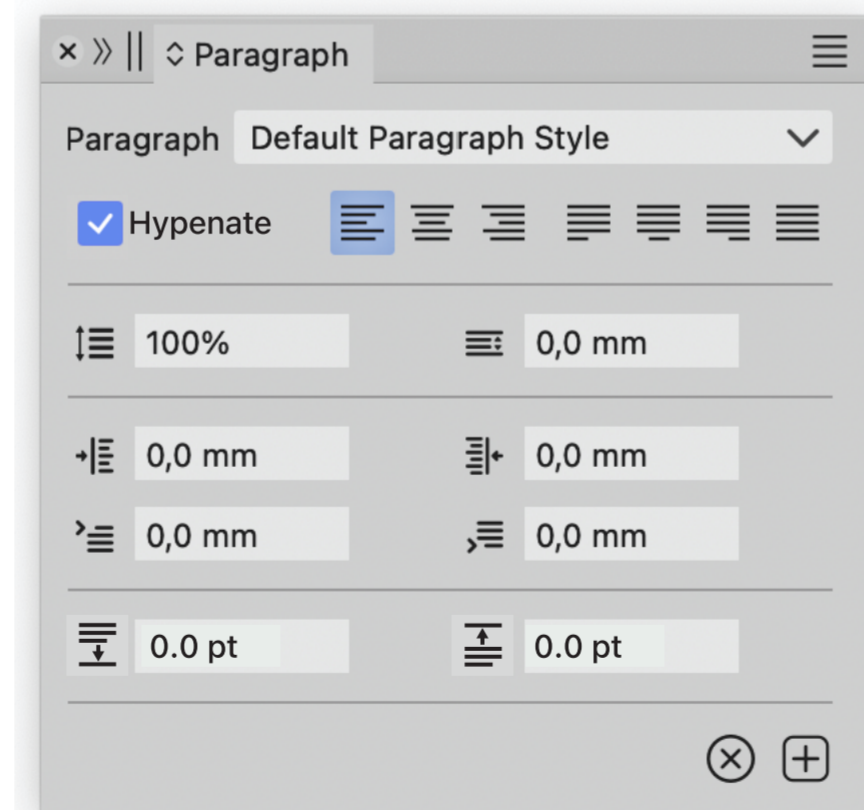
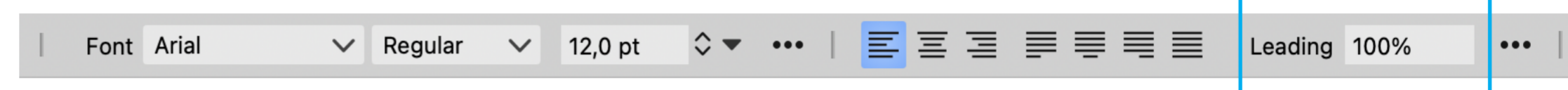
4. Leading as an Character attribute

The InDesign/Illustrator approach where leading is a character attribute.

Which means that you can apply more than one leading value within the same paragraph. The largest leading value in a line of type determines the leading for that line.

In preferences or options you could indicate whether the leading applies to the entire paragraph, instead of to text within a paragraph.

The default percentage leading should also have a setting somewhere at program or document level.



Leading Percentage 100%

Leading % 100%

Extra
Fixed
Percentage (Default)

The question is whether leading selector should also be visible in the context bar or simply follow the setting in the panel with the corresponding value selector.

+

=

%

Extra
Fixed
Percentage (Default)

Leading methods
Selection for leading mode: percentage (relative), Fixed (absolute pt), or Extra (added pt).

Percentage	Fixed (pt)	Extra (pt)
0%	4.0	-10.0
10%	6.0	-9.0
20%	8.0	-8.0
30%	8.0	-7.0
40%	9.0	-6.0
50%	10.0	-5.0
60%	11.0	-4.0
70%	✓ 12.0	-3.0
80%	14.0	-2.0
90%	16.0	-1.0
✓ 100%	20.0	✓ 0.0
110%	22.0	+1.0
120%	24.0	+2.0
130%	36.0	+3.0
140%	48.0	+4.0
150%	60.0	+5.0
160%	72.0	+6.0
170%	96.0	+7.0
180%	144.0	8+.0
190%	288.0	9+.0
200%		+10.0

Note!
- The default percentage leading should have a setting at app or document level.

Relative and Extra based on font metrics ascender and descender, not the so-called 'default line-height' which is basically a font designer's personal preference which is irrelevant. Leading is highly dependent on line width and application.

5. Out of the box, Leading as an text attribute

Drop the character and paragraph panel and create a new panel with the name 'Text'. All text and paragraph formatting specifications conveniently together. Leading as a character attribute. This direction is more similar to Amadine and Freehand approach.

On the one hand simplified and more intuitive, on the other hand **enriched with the option to specify font size in multiple ways**. Since Vectorstyler is primarily a drawing program, a very versatile solution for working creatively with text.

My favorite

The image shows the 'Text' panel in Vectorstyler with various leading options. A callout box at the top right explains the new panel menu structure. Three callout boxes on the left describe the 'Size' and 'Leading' options. Three callout boxes on the right show the available leading values for different units: Percentage, Fixed (pt), and Extra (pt).

New Panel Menu, merged from the old character and paragraph, reorganized and simplified.

- Save Text Style...
- Save Text Preset...
- ✓ Apply Leading to Paragraph
-
-
- Break Style Link
- Clear Text Overrides

Size options:

- Char Height (default)
- Cap Height
- x Height

Leading options:

- Extra
- Fixed
- Percentage

Percentage leading values:

- 5.0
- 6.0
- 7.0
- 8.0
- 9.0
- ✓ 10.0
- 11.0
- 12.0
- 14.0
- 16.0
- 18.0
- 20.0
- 22.0
- 24.0
- 36.0
- 48.0
- 60.0
- 72.0

Fixed (pt) leading values:

- 4.0
- 6.0
- 8.0
- 8.0
- 9.0
- 10.0
- 11.0
- ✓ 12.0
- 14.0
- 16.0
- 20.0
- 22.0
- 24.0
- 36.0
- 48.0
- 60.0
- 72.0
- 96.0
- 144.0
- 288.0

Extra (pt) leading values:

- 10.0
- 9.0
- 8.0
- 7.0
- 6.0
- 5.0
- 4.0
- 3.0
- 2.0
- 1.0
- ✓ 0.0
- +1.0
- +2.0
- +3.0
- +4.0
- +5.0
- +6.0
- +7.0
- 8+.0
- 9+.0
- +10.0