

Rich Text API Details

This page discusses some of the details of the current Rich Text API prototype.

The current prototype adds **one** (not ~~two~~) new node to the scene layer.

TextFlow, subclass of Parent. Includes the following properties:

name	type
wrappingWidth	double
textAlignment	TextAlignment

Notes:

1. Font can be set in CSS, even though it doesn't exist as property in the node, and it is inherit to the children.
2. Font-smoothing, underline, strikethrough, fill won't work the same way since they are not inherit in CSS.
3. The fact that TextFlow does not have a font makes tab expansion wrong (tab is 8 whitespace in which font?), a possible solution for this problem is to added a tab stop list to TextFlow.
4. TextFlow does not have a BoundsType property, it is always logic. Laying out the children using visual bounds doesn't seen to make sense.
5. Line spacing would be easy to add (See [RT-21683](#)).
6. Consider change TextFlow to subclass Region instead of Parent **New: most likely doing this!**
7. What is the best way to handle TextFlow nested in TextFlow?
8. **Properties removed (relative to the original proposal): x, y, textOrigin**

Alternative names for TextFlow (**New: very unlikely the name will change**):
Paragraph, TextGroup, TextPane, TextBlock, DIV, TextLayoutPanel, RichTextPane.

Instead of adding a new node (e.i ~~Span~~) we decided to use the existent Text node:
When Text is a child of TextFlow only the following properties are respected:

name	type
text	String
font	Font
fontSmoothingType	FontSmoothingType
underline	boolean
strikethrough	boolean

Plus all the properties in the super classes (Shape & Node)

Note that some of the properties in Text are **ignored** when it is a child of TextFlow. They are:
x, y, textOrigin, boundsType, wrappingWidth, textAlignment.

Using these new elements a rich text layout can be accomplished using code or markup (FXML), and entirely styled using properties or CSS.
See some examples in [Rich Text API Samples](#).

Relavants FXML bugs for Rich Text:

1. [RT-24466](#) - support load object hierarchy from string
2. [RT-24336](#) - better whitespace handling in the text node