

# 多源输入的富文本渲染实现

刘丰恺（好好学习）

# 个人简介

刘丰恺

- 头条研发 好好学习客户端 致力于 Flutter 相关探索、场景化开发
- 曾就职阿里、dashbase，从事后端 infra 相关框架开发
- GitHub: <https://github.com/lfkdsk>

# 背景

## 遇到的问题

1. WebView、Native 混合

2. Quill Delta 格式不支持

3. 支持多种语法输入



# 背景

## 方案 Feature List

1. Parser 拓展便利、易于支持多种输入源
2. 可拓展性强、组件订制灵活
3. 前后端分离、解耦，降低开发使用成本

# 演示

下午5:44 0.71K/s ⚡ Bug

## Techo 富文本演示

### 简介

Techo 富文本渲染组件

- 使用独有的多输入源进行解析 rich\_syntax
- 支持多数 quill clipboard 支持的 html 转换
- 提供 datus 对富文本数据结构进行支持
- 解析、渲染，前后端分离，使用 delta 数据结构进行数据连接

### 基本用法

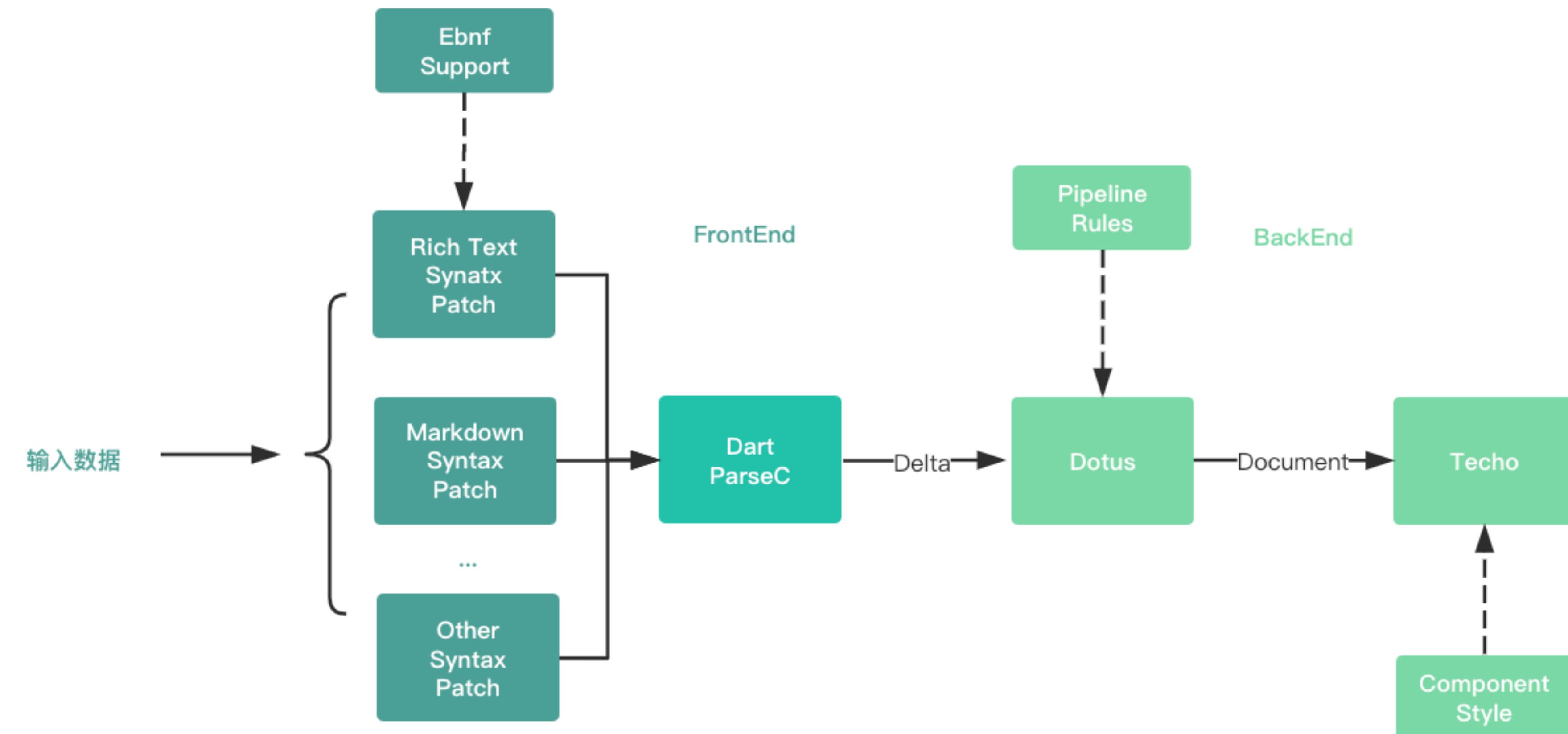
[https://code.byted.org/learning\\_client/Techo](https://code/byted.org/learning_client/Techo)

### 输入样例

```
<body>
<!-- Test Heading -->
<h1>Heading</h1>
<h1>H1</h1>
<h2>H2</h2>
<h3>H3</h3>

<h1>Paragraph</h1>
<!--文本段-->
<p>换行 <br> <strong>加粗</strong> <i>斜体</i></p>
<!--链接-->
<p>
  <a class="pgc-link"
    href="http://toutiao.com/group/6715326135770546695/?iid=0&app=news_article"
    target="_blank">文章</a> Ifkdsk <i> i </i>
</p>
<!--序列-->
<h1>List</h1>
<ol>
  <li>Order Item 0</li>
  <li>Order Item 1</li>
  <li>Order Item 2</li>
<ul>
  <li>UnOrder Item 0</li>
  <li>UnOrder Item 1</li>
```

# 结构图



# Front End



# Dart ParseC ToolChain

## 1. ScanLess

## 2. Parser Combinator

## 3. Error Recovery

```
/// attr item (a | a=' ' | a="")
Parser attr() =>
    attrName() &
    (ref(EQUAL) &
        (bracket('""', regex('[^"]*')) |
         bracket("'''", regex("[^']*")) |
         word().star()))
    .optional();

/// <tag-name attr-list (/>)
Parser startTag() =>
    (ref(START_PAREN) & ref(tagName) & ref(attr).star()).permute([1,
2]);
Parser<String> closeTag() =>
    (ref(symbol, '</') & tagName() & END_PAREN()).flatten().trim();

/// inner html
/// 1. multi-tag
/// 2. <xx> { text } </xx> lazy expect '</'
Parser innerTag() =>
    string('<br>').map((text) => EmptyTag.empty('br')) |
    ref(multiTag) |
    regex('[^&<]*').flatten().map((text) {
        if (text.contains('{!--') && text.contains('--!}') ) {
            return ExtraTag(text);
        }
        return TextLeaf(text);
    });

/// 1.<tag-name attr-list />
/// 2.<tag-name attr-list> { inner tags } </tag-name>
Parser multiTag() =>
    ref(startTag) &
    (ref(CLOSE_PAREN) |
        (END_PAREN() & ref(innerTag).star() & closeTag()) |
        ref(END_PAREN));
```

# ParseC Example

1.



```
Parser<String> parser = regex('[a-z]');
```

2.



```
final parserLetter = regex('[a-z]');
final parserNumber = range(0, 9);
final parserID = parserA & parserB;
final numberORLetter = parserNumber | parserLetter;
```

3.

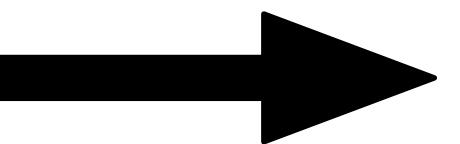


```
final parserLetter = regex('[a-z]');
final parserNumber = range(0, 9);
// ([a-z] | [0-9])+
final parserID = (parserA | parserB).plus();
// ([a-z] | [0-9])*
final parserID = (parserA | parserB).star();
```

# RichText Syntax Rules

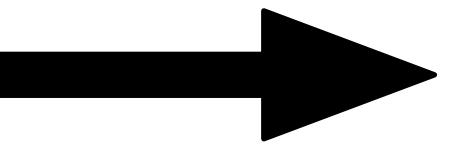
## Examples

```
● ● ●  
<body>  
  <h1>Heading One! </h1>  
</body>
```



insert< Heading >  
insert< ↴ > + {heading: 1}

```
● ● ●  
<body>  
  <p> 换行 <br> <strong>加粗</strong> <i>斜体</i></p>  
</body>
```

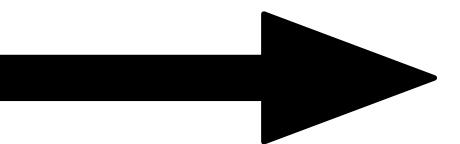


insert< 换行 ↴ >  
insert< 加粗 > + {bold: true}  
insert< 斜体 > + {italic: true}  
insert< ↴ >

# RichText Syntax Rules

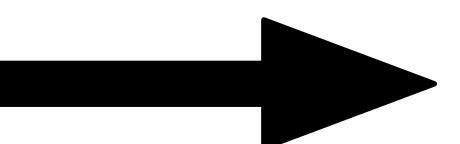
## Examples

```
● ● ●  
  
<body>  
  <ol>  
    <li>Order Item 0</li>  
    <li>Order Item 1</li>  
    <ul>  
      <li>UnOrder Item 0</li>  
      <li>UnOrder Item 1</li>  
    </ul>  
  </ol>  
</body>
```



```
insert< Order Item 0 >  
insert< ↴ + {intend: 0, block: ol}>  
  
insert< Order Item 1 >  
insert< ↴ + {intend: 0, block: ol}>  
  
insert< UnOrder Item 0 >  
insert< ↴ + {intend: 1, block: ul}>  
  
insert< UnOrder Item 1 >  
insert< ↴ + {intend: 1, block: ul}>
```

```
● ● ●  
  
<body>  
  <p>  
      
  </p>  
</body>
```



```
insert< > + {embed:  
  {source: <url>,  
   type: image}  
}
```

# RichText Syntax

```
● ● ●  
<!-- Test Heading --&gt;<h1>Heading</h1>

# H1</h1>H2</h2>H3</h3>Paragraph</h1><!--文本段--&gt;<p>换行 <br> <strong>加粗</strong> <i>斜体</i></p><!--链接--&gt;<p><a class="pgc-link" href="http://toutiao.com/group/6715326135770546695/?iid=0&amp;app=news_article" target="_blank">文章</a> lfkdsk <i> i </i> </p><!--序列--&gt;<h1>List</h1> - Order Item 0 - Order Item 1 - Order Item 2 - UnOrder Item 0 - UnOrder Item 1 ``` import 'package:flutter/material.dart'; ``` ``` import 'package:techo/techo.dart'; ``` ``` void main() { ``` ``` runApp(TechoApp()); ``` ``` } ``` Image</h1> <p></p> <p></p>


```

```
insert( Heading )  
insert( ↵ ) + {heading: 1}  
insert( H1 )  
insert( ↵ ) + {heading: 1}  
insert( H2 )  
insert( ↵ ) + {heading: 2}  
insert( H3 )  
insert( ↵ ) + {heading: 3}  
insert( Paragraph )  
insert( ↵ ) + {heading: 1}  
insert( 换行 ↵ )  
insert( 加粗 ) + {bold: true}  
insert( 斜体 ) + {italic: true}  
insert( ↵ )  
insert( 文章 ) + {class: pgc-link, href: http://toutiao.com/group/6715326135770546695/?  
iid=0&amp;app=news_article, target: _blank, link: http://toutiao.com/group/6715326135770546695/?  
iid=0&amp;app=news_article}  
insert( lfkdsk )  
insert( i ) + {italic: true}  
insert( ↵List )  
insert( ↵ ) + {heading: 1}  
insert( Order Item 0 )  
insert( ↵ ) + {intend: 0, block: ol}  
insert( Order Item 1 )  
insert( ↵ ) + {intend: 0, block: ol}  
insert( Order Item 2 )  
insert( ↵ ) + {intend: 0, block: ol}  
insert( UnOrder Item 0 )  
insert( ↵ ) + {intend: 1, block: ul}  
insert( UnOrder Item 1 )  
insert( ↵ ) + {intend: 1, block: ul}  
insert( Code Block )  
insert( ↵ ) + {heading: 1}  
insert( import 'package:flutter/material.dart'; )  
insert( ↵ ) + {block: code}  
insert( import 'package:techo/techo.dart'; )  
insert( ↵ ) + {block: code}  
insert( void main() { )  
insert( ↵ ) + {block: code}  
insert( runApp(TechoApp()); )  
insert( ↵ ) + {block: code}  
insert( } )  
insert( ↵ ) + {block: code}  
insert( Image )  
insert( ↵ ) + {heading: 1}  
insert( Small Image )  
insert( ↵ ) + {heading: 2}  
insert( • ) + {embed: {source:  
https://www.gstatic.com/images/branding/googlelogo/2x/googlelogo_color_284x96dp.png, type: image}}  
insert( ↵Long Image )  
insert( ↵ ) + {heading: 2}  
insert( • ) + {embed: {source: http://sf1-ttcdn-tos.pstatp.com/img/learning/7b9cd6bda1e07a425d73d0dd94c5aed~1125x0.jpg, type: image}}  
insert( ↵ )
```

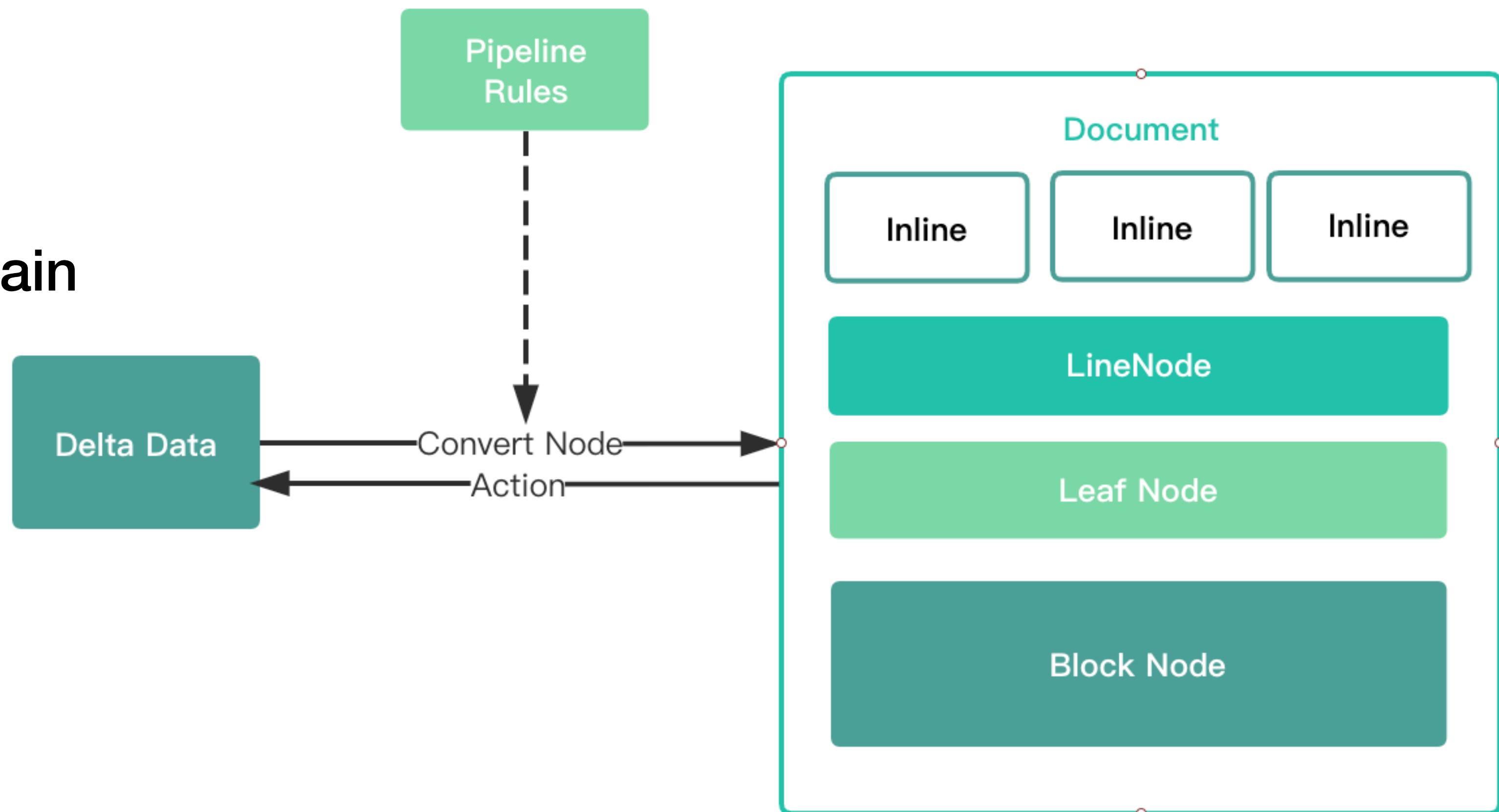
# BackEnd



## 1. Delta <=> Nodes

## 2. Actions: insert, delete, retain

## 3. Pipeline Rules



# Dotus

## Examples

```
insert< Heading >  
insert< ↴ > + {heading: 1}
```



```
¶ <Heading> ↴ {heading: 1}
```

```
insert< 换行 ↴ >  
insert< 加粗 > + {bold: true}  
insert< 斜体 > + {italic: true}  
insert< ↴ >
```



```
¶ <换行> ↴  
¶ <加粗> bold → <斜体> italic ↴
```

```
insert< Order Item 0 >  
insert< ↴ > + {intend: 0, block: ol}  
insert< Order Item 1 >  
insert< ↴ > + {intend: 0, block: ol}  
insert< Order Item 2 >  
insert< ↴ > + {intend: 0, block: ol}
```



§ {ol}

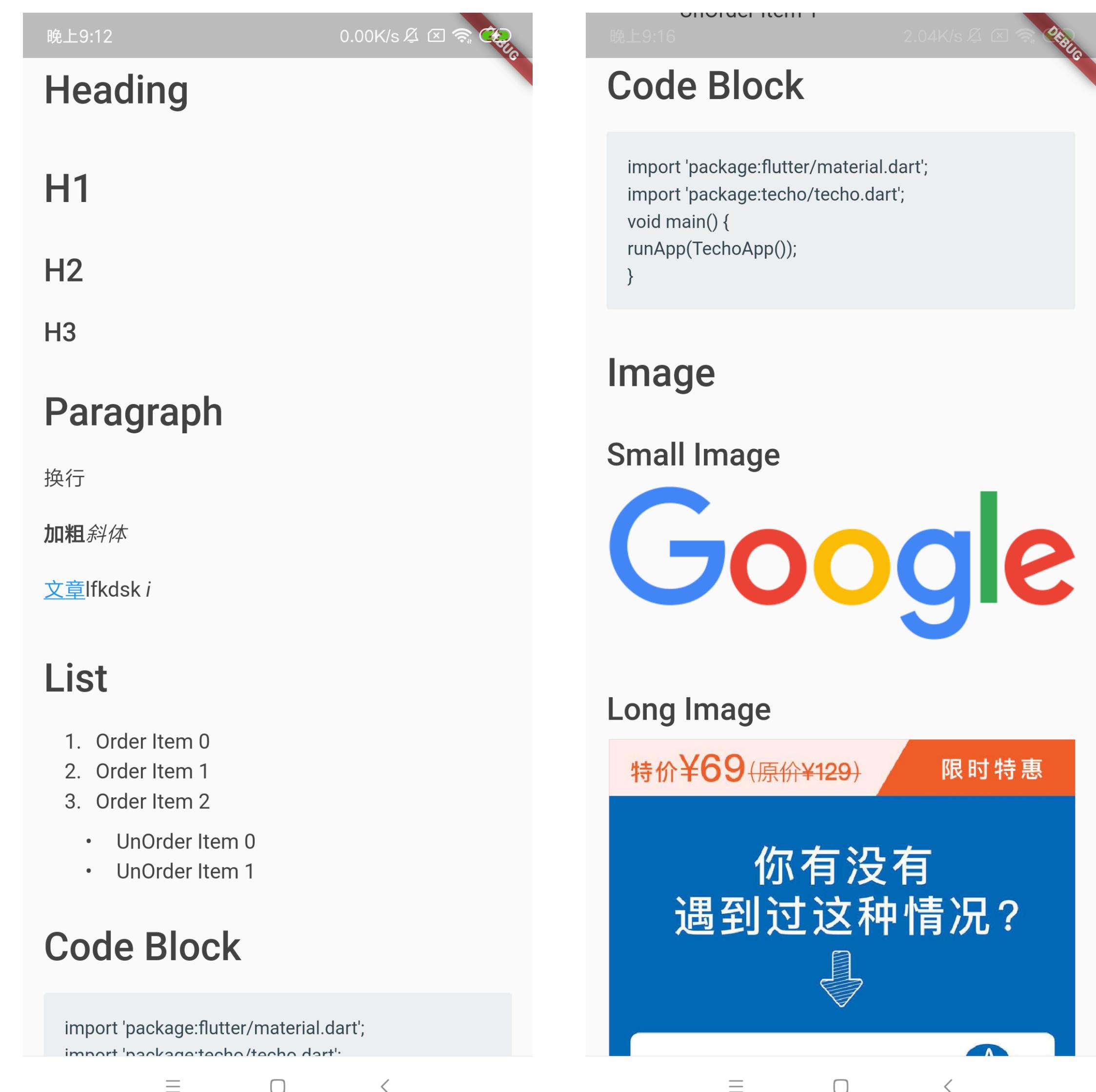
```
  └ ¶ <Order Item 0> ↴ {intend: 0, block: ol}  
  └ ¶ <Order Item 1> ↴ {intend: 0, block: ol}  
  └ ¶ <Order Item 2> ↴ {intend: 0, block: ol}
```

# Techo Viewer

## 1. 组件对应

## 2. 样式支持

## 3. Embed 组件拓展支持



# For Future



# 后续进展

1. Editable Support.
2. CSS Support.
3. Script To Widgets.



THANKS.



好好学习 Flutter 相关建设进展