Out-of-flow Positioned Fragmentation

Alison Maher
almaher@microsoft.com

Fragmentation Overview

Definitions

- Fragmentainer
  - Fragment container
  - A box that contains a portion of fragmented content

Definitions

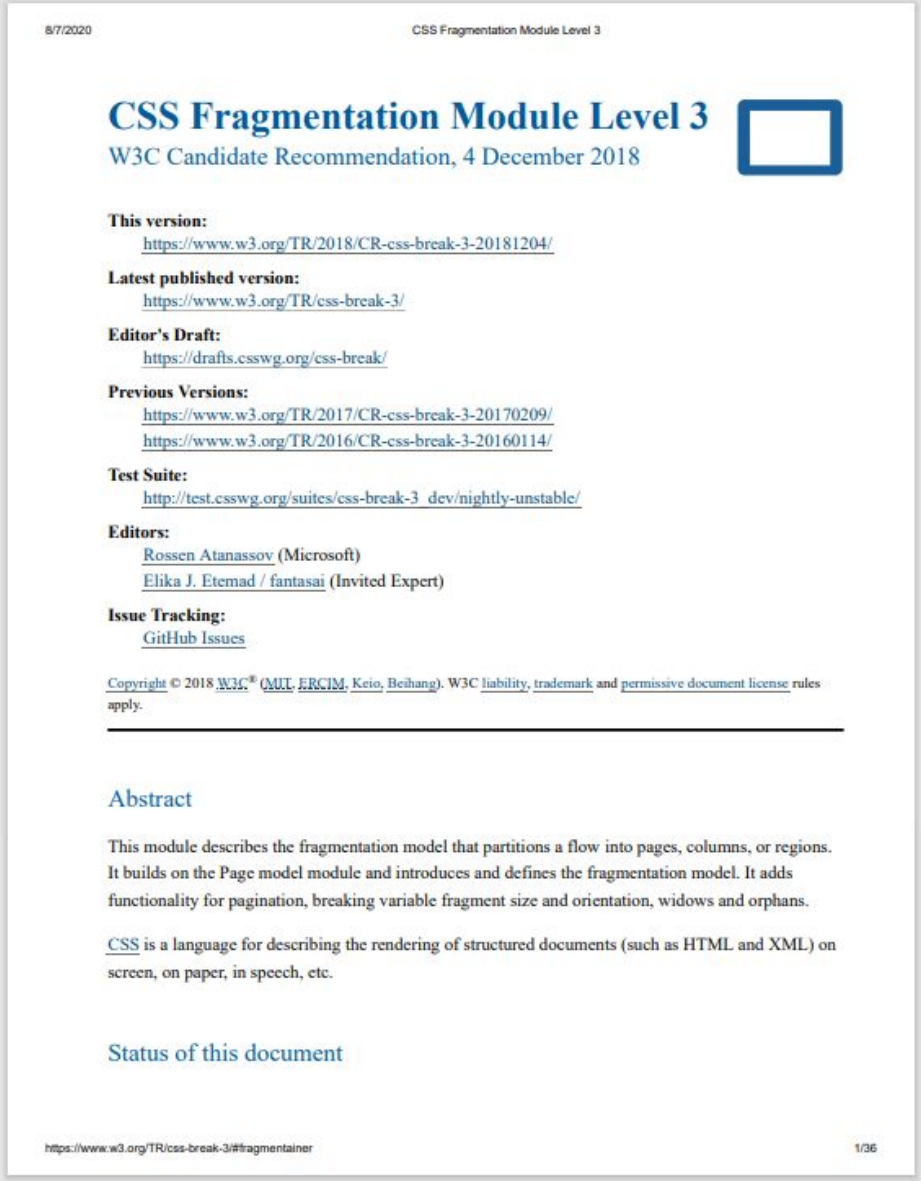
- Fragmentainer
  - Fragment container
  - A box that contains a portion of fragmented content
- Fragmentation contexts
  - Collection of fragmentainers

Definitions

- Fragmentainer
  - Fragment container
  - A box that contains a portion of fragmented content
- Fragmentation contexts
  - Collection of fragmentainers
- Fragmentation
  - “The process of splitting a content flow across the fragmentainers that form a fragmentation context.”

Paged Media

- Fragmentainer = page
- Fragmentation context = Collection of pages



8/7/2020 CSS Fragmentation Module Level 3

CSS Fragmentation Module Level 3

W3C Candidate Recommendation, 4 December 2018

This version:
https://www.w3.org/TR/2018/CR-css-break-3-20181204/

Latest published version:
https://www.w3.org/TR/css-break-3/

Editor's Draft:
https://drafts.csswg.org/css-break/

Previous Versions:
https://www.w3.org/TR/2017/CR-css-break-3-20170209/
https://www.w3.org/TR/2016/CR-css-break-3-20160114/

Test Suite:
http://test.csswg.org/suites/css-break-3_dev/nightly-unstable/

Editors:
Rossen Atanassov (Microsoft)
Elika J. Etemad / fantasai (Invited Expert)

Issue Tracking:
GitHub Issues

Copyright © 2018 W3C® (MIT, ERCIM, Keio, Beihang). W3C liability, trademark and permissive document license rules apply.

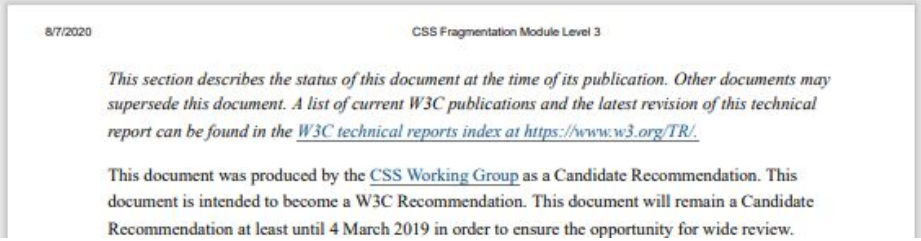
Abstract

This module describes the fragmentation model that partitions a flow into pages, columns, or regions. It builds on the Page model module and introduces and defines the fragmentation model. It adds functionality for pagination, breaking variable fragment size and orientation, widows and orphans.

CSS is a language for describing the rendering of structured documents (such as HTML and XML) on screen, on paper, in speech, etc.

Status of this document

https://www.w3.org/TR/css-break-3/#fragmentainer 1/36



8/7/2020 CSS Fragmentation Module Level 3

This section describes the status of this document at the time of its publication. Other documents may supersede this document. A list of current W3C publications and the latest revision of this technical report can be found in the W3C technical reports index at https://www.w3.org/TR/.

This document was produced by the CSS Working Group as a Candidate Recommendation. This document is intended to become a W3C Recommendation. This document will remain a Candidate Recommendation at least until 4 March 2019 in order to ensure the opportunity for wide review.

Multi-columns

- Fragmentainer = column
- Fragmentation context = Collection of columns

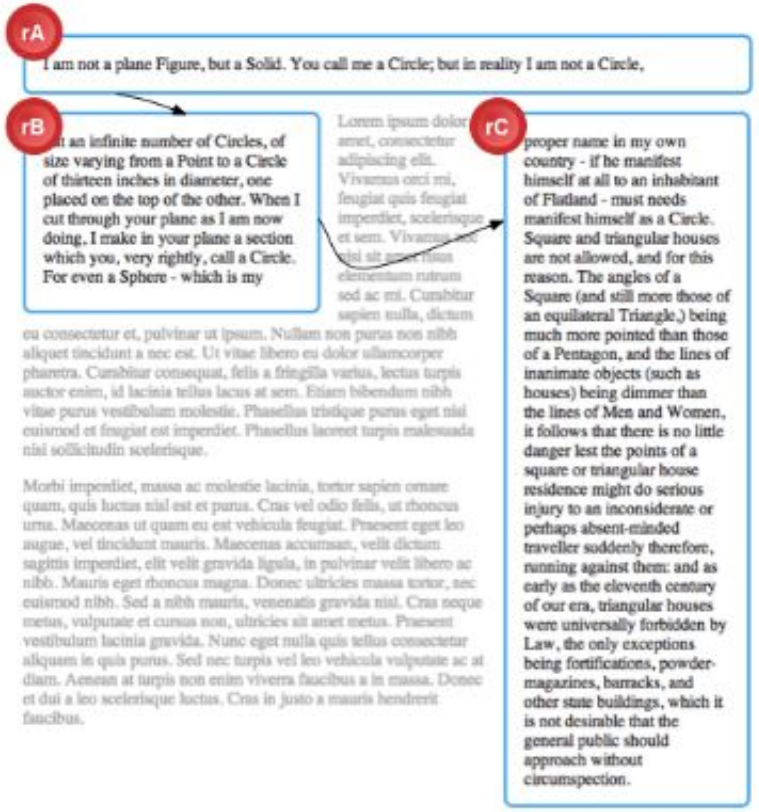
Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper

suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui

blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum.

Regions

- Fragmentainer = region
- Fragmentation context = Collection of regions

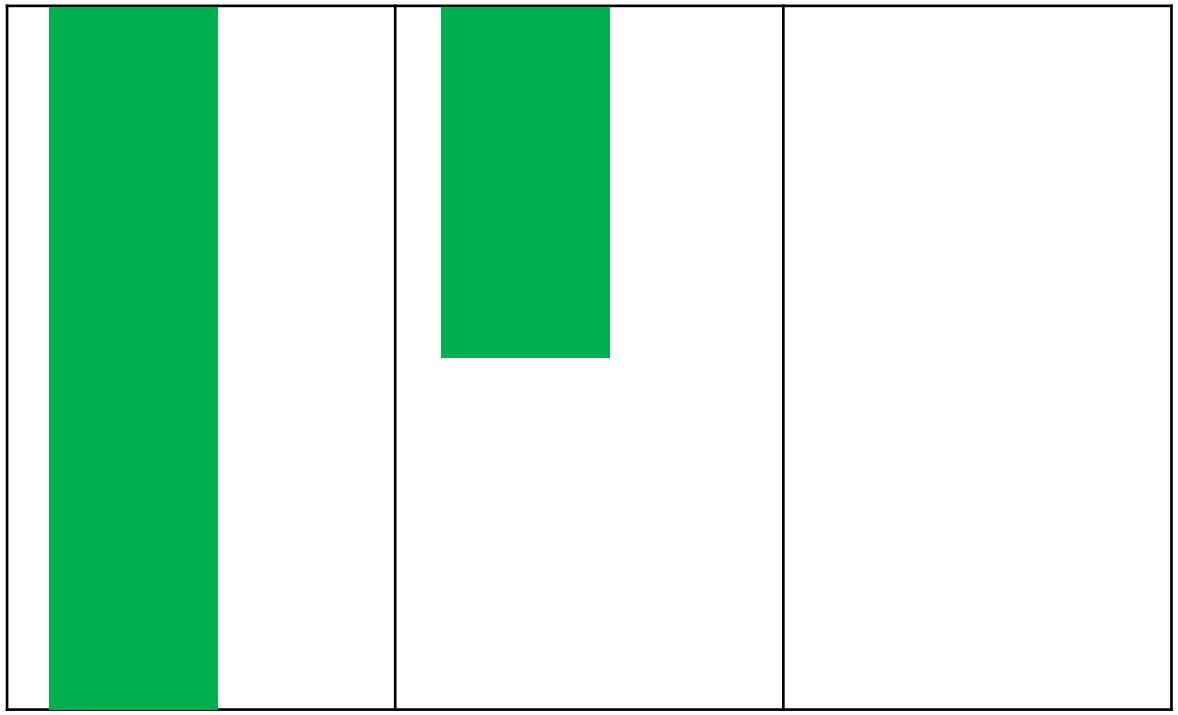


Definitions

- Fragment
  - “The portion of a box that belongs to exactly one fragmentainer.”

Definitions

- Fragment
  - "The portion of a box that belongs to exactly one fragmentainer."



Definitions

- Fragment
  - “The portion of a box that belongs to exactly one fragmentainer.”
- Monolithic content
  - Content that isn’t allowed to fragment
  - Ex: images and videos

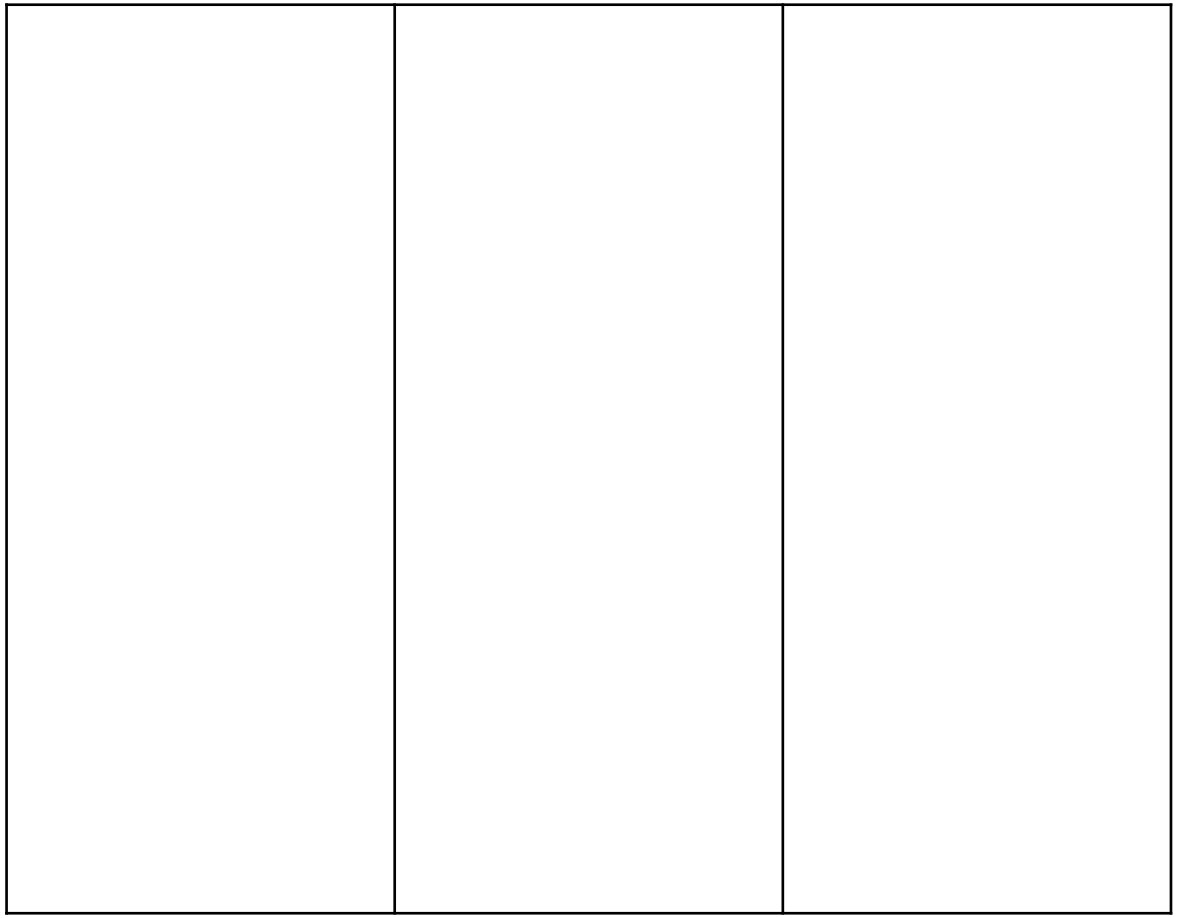
CSS Specifications

- CSS Fragmentation Module Level 3 (https://drafts.csswg.org/css-break/)
- CSS Multi-column Layout Module Level 1 (https://drafts.csswg.org/css-multicol/)
- CSS Paged Media Module Level 3 (https://drafts.csswg.org/css-page-3/)
- CSS Regions Module Level 1 (https://drafts.csswg.org/css-regions/)

Fragmentation in LayoutNG

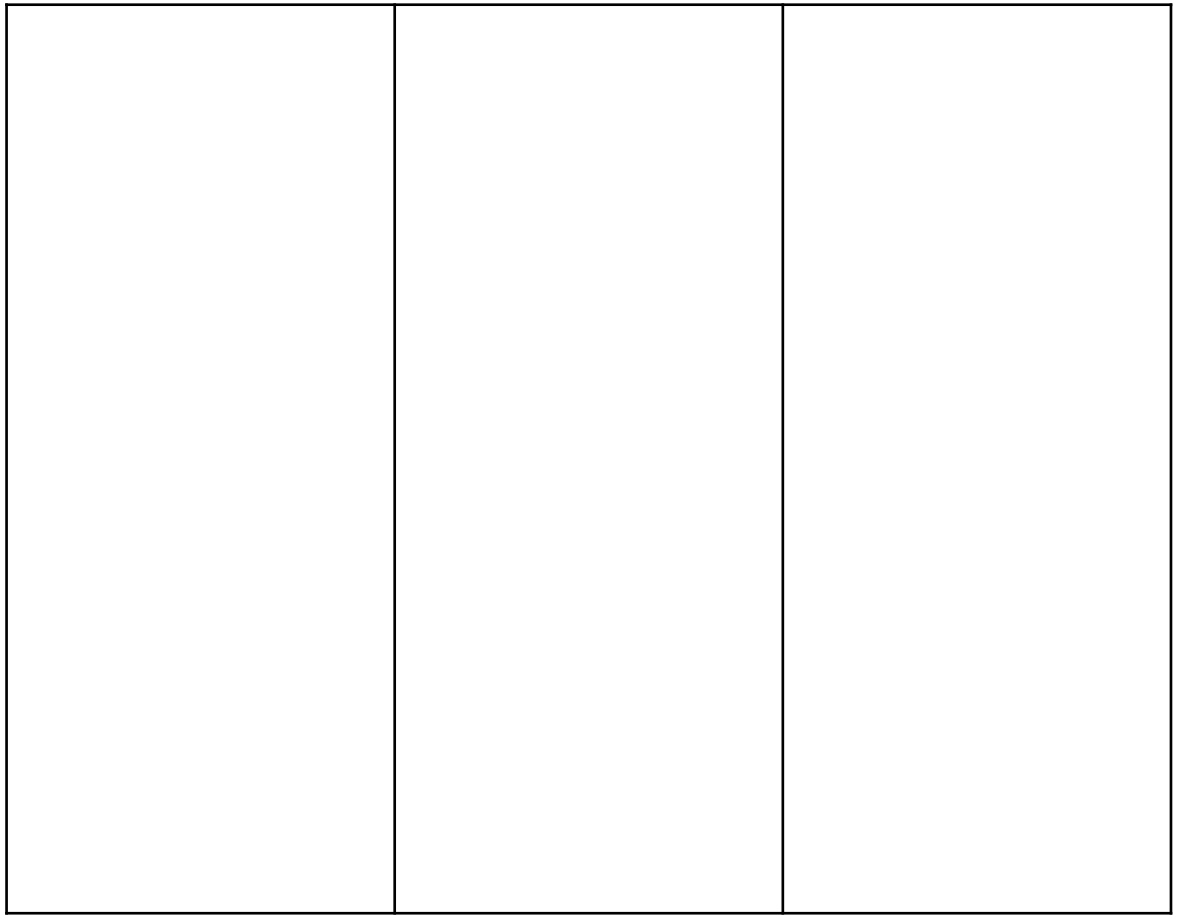
Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}
```



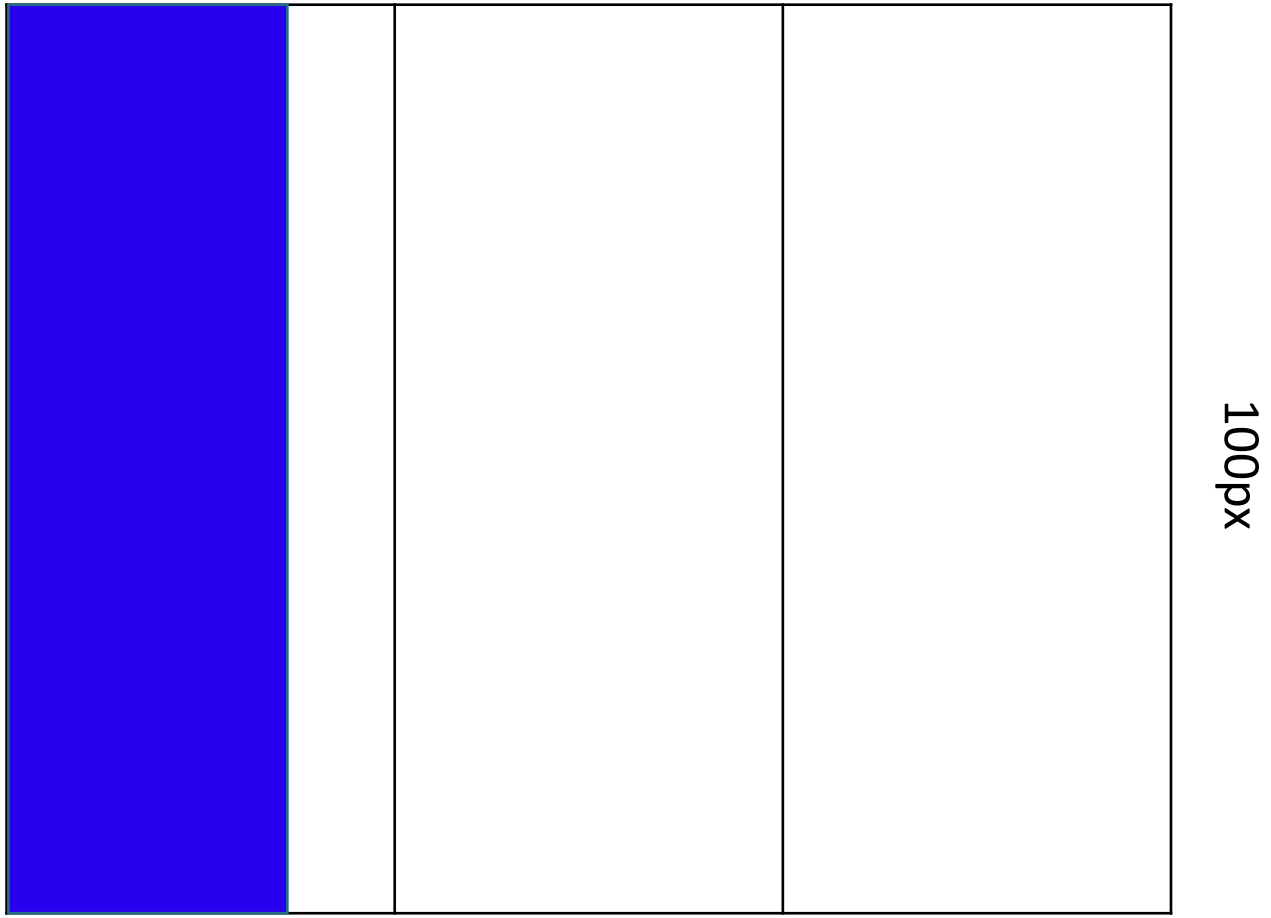
Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}

#child {
  height: 250px;
  background: blue;
}
```



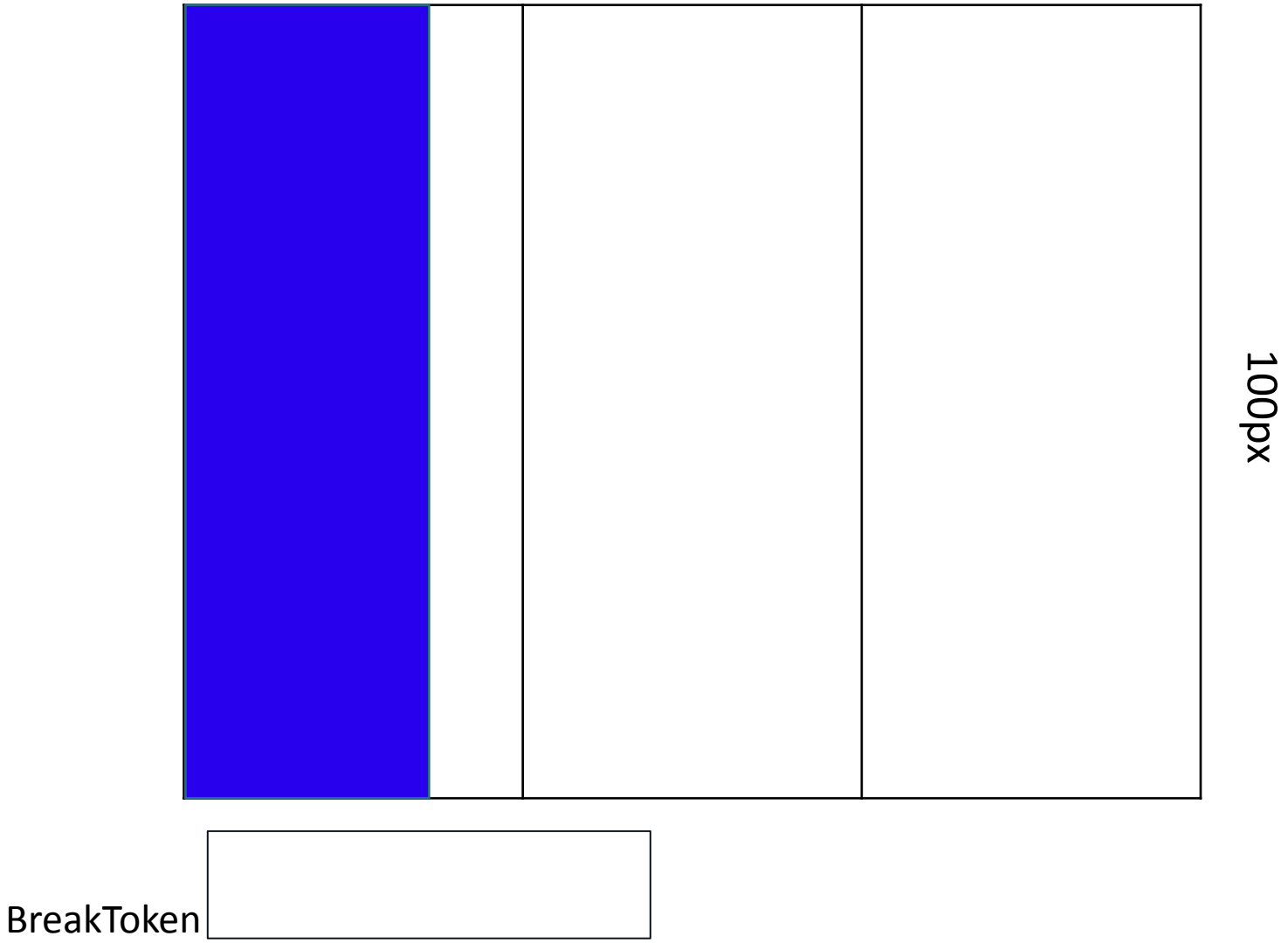
Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}

#child {
  height: 250px;
  background: blue;
}
```



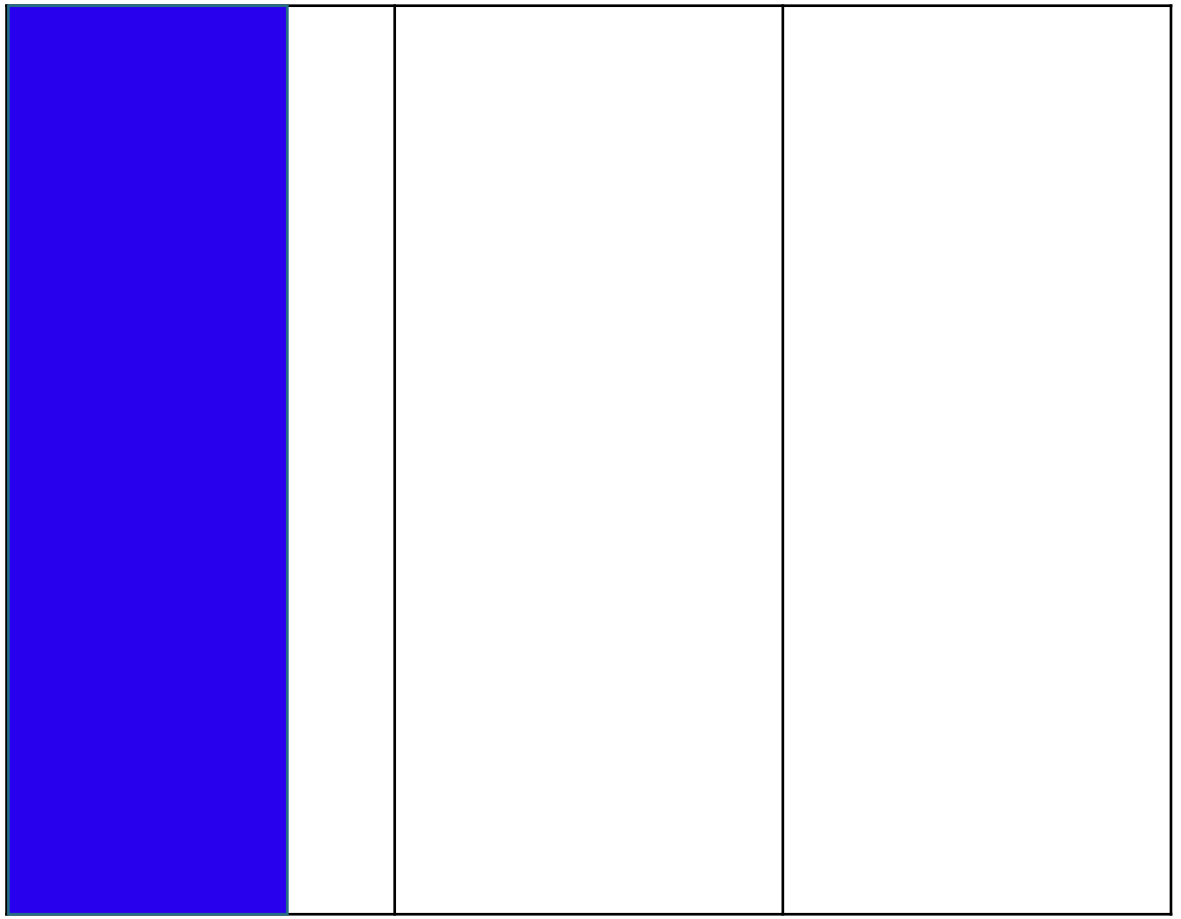
Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}

#child {
  height: 250px;
  background: blue;
}
```



Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}

#child {
  height: 250px;
  background: blue;
}
```

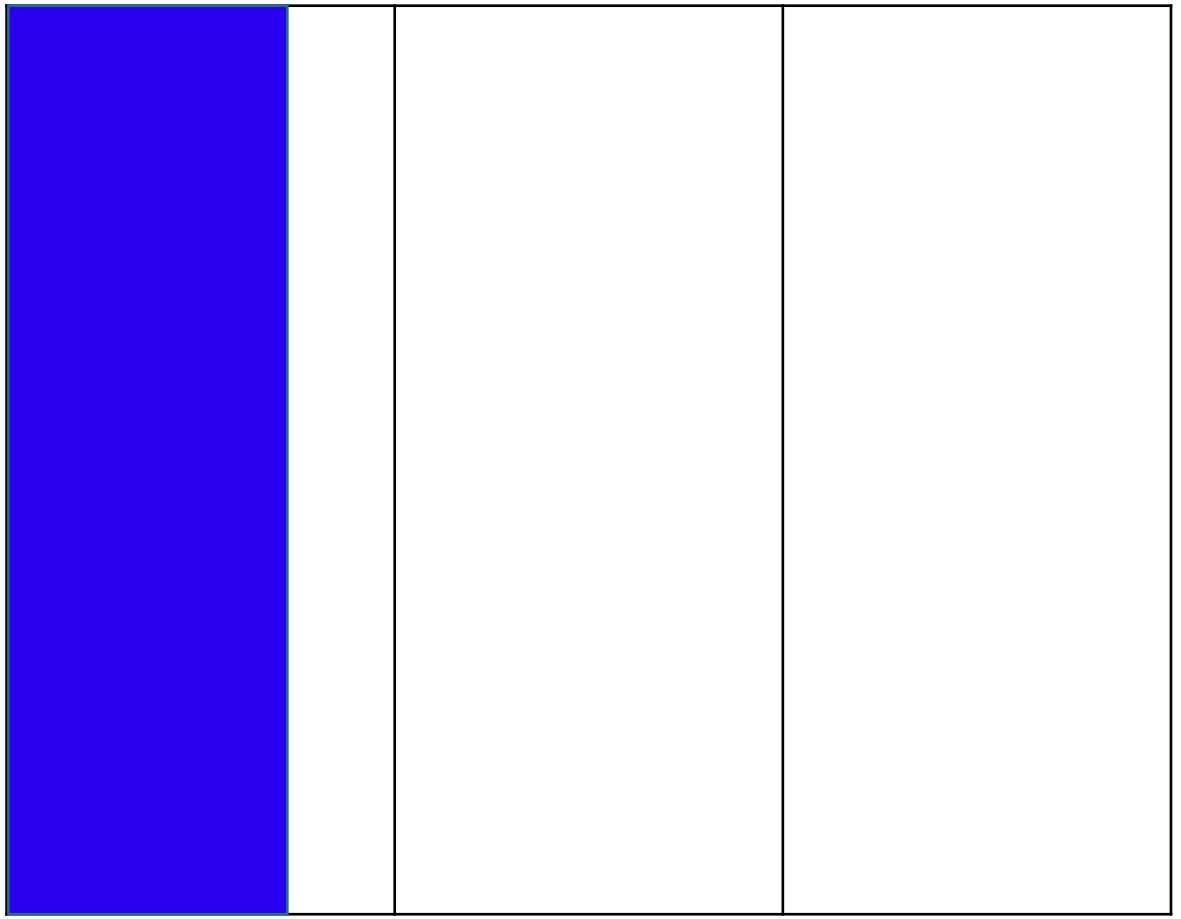


BreakToken

consumed_block_size

Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}

#child {
  height: 250px;
  background: blue;
}
```

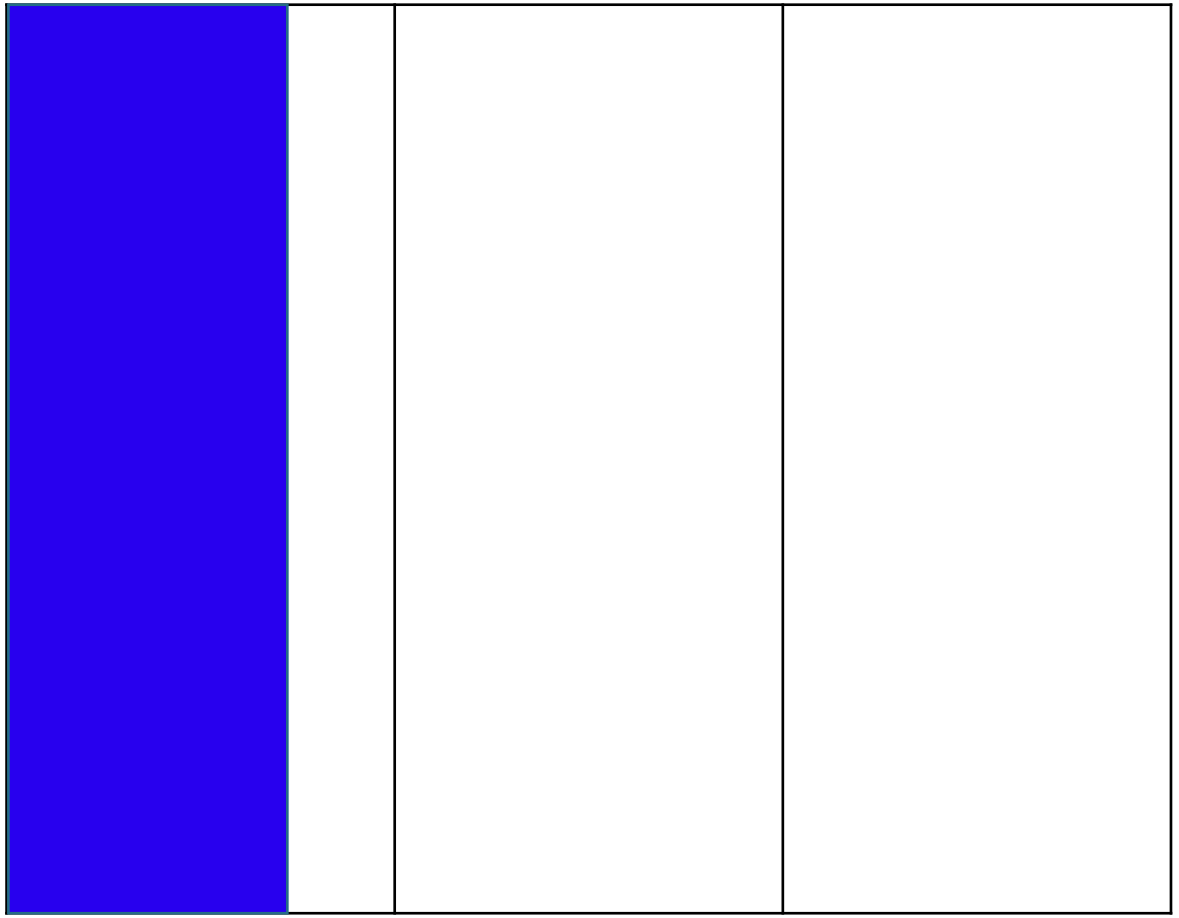


BreakToken

consumed_block_size=100

Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}

#child {
  height: 250px;
  background: blue;
}
```

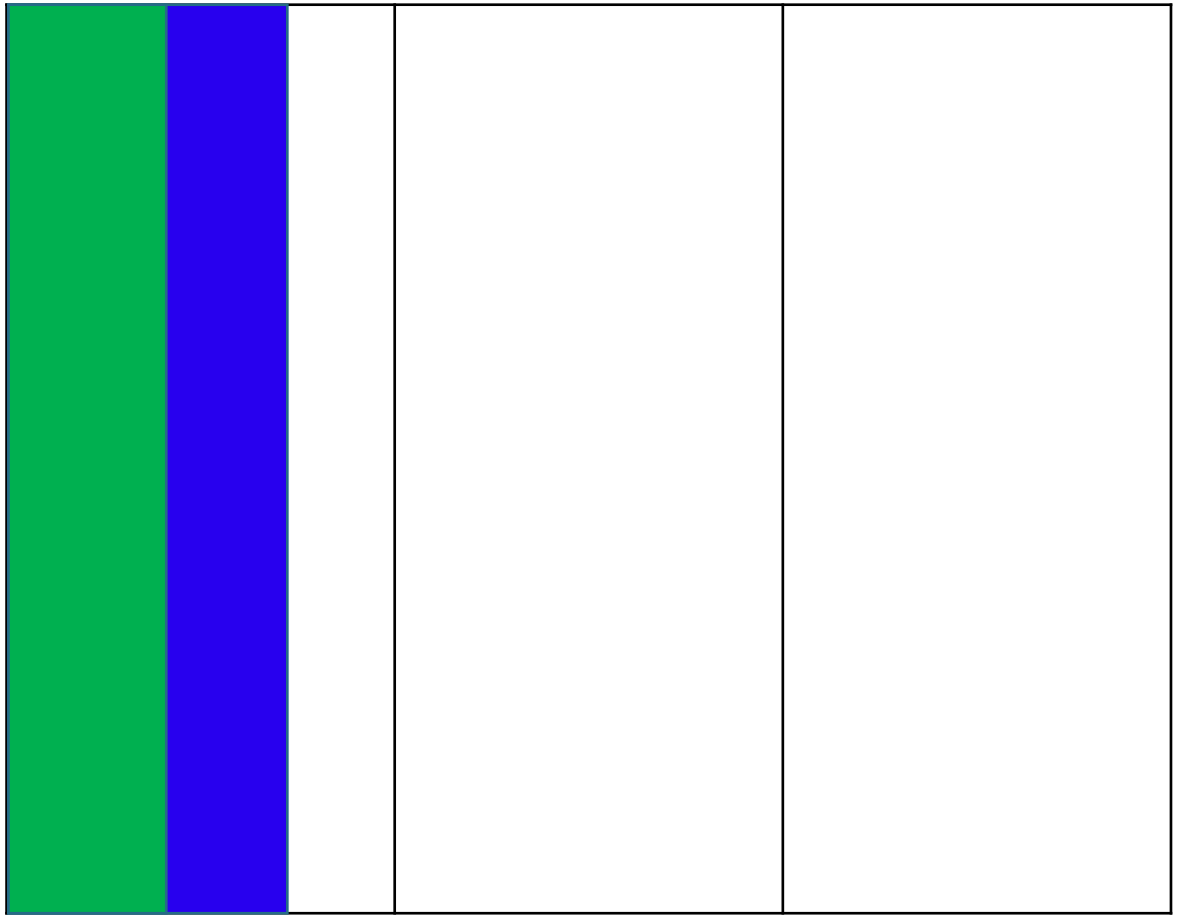


BreakToken

consumed_block_size=100
child_break_tokens

Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}

#child {
  height: 250px;
  background: blue;
}
```

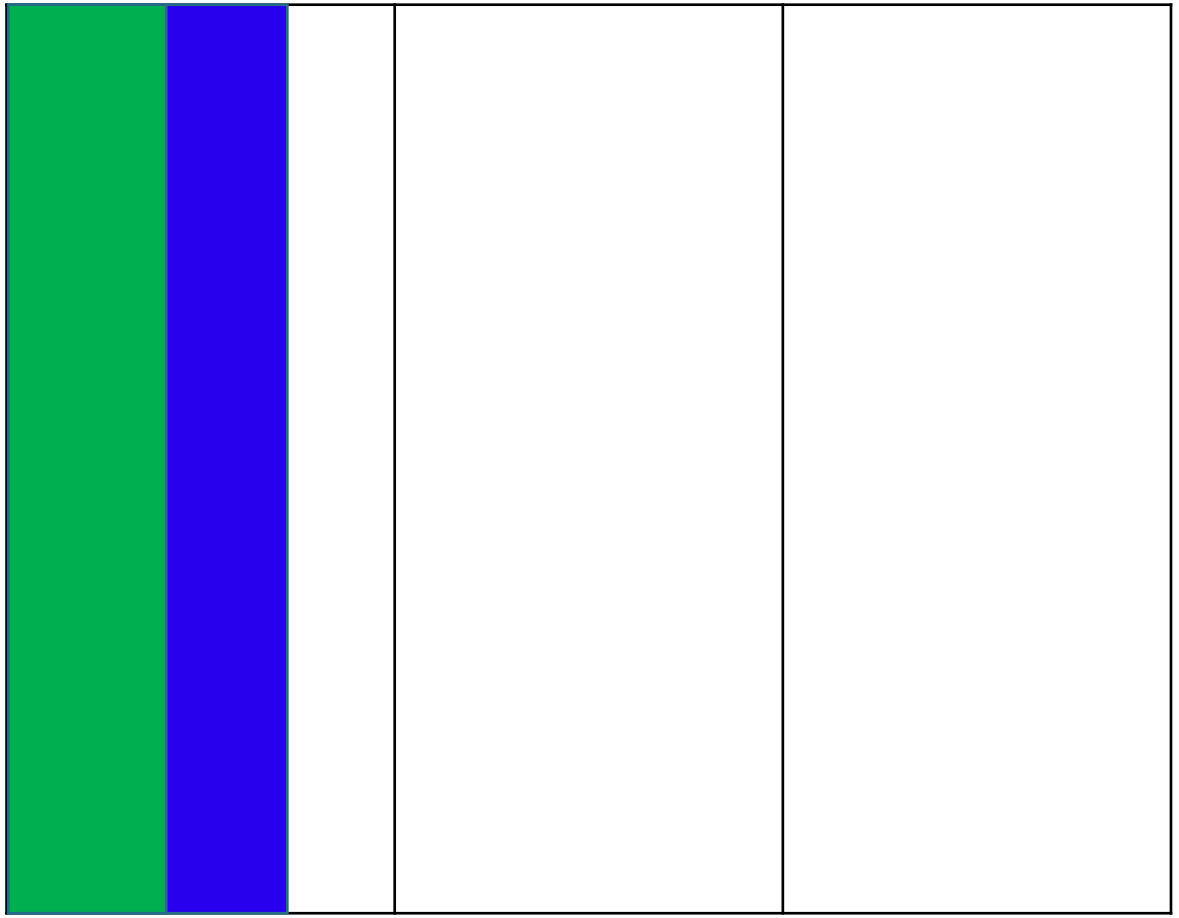


BreakToken

consumed_block_size=100
child_break_tokens

Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}

#child {
  height: 250px;
  background: blue;
}
```

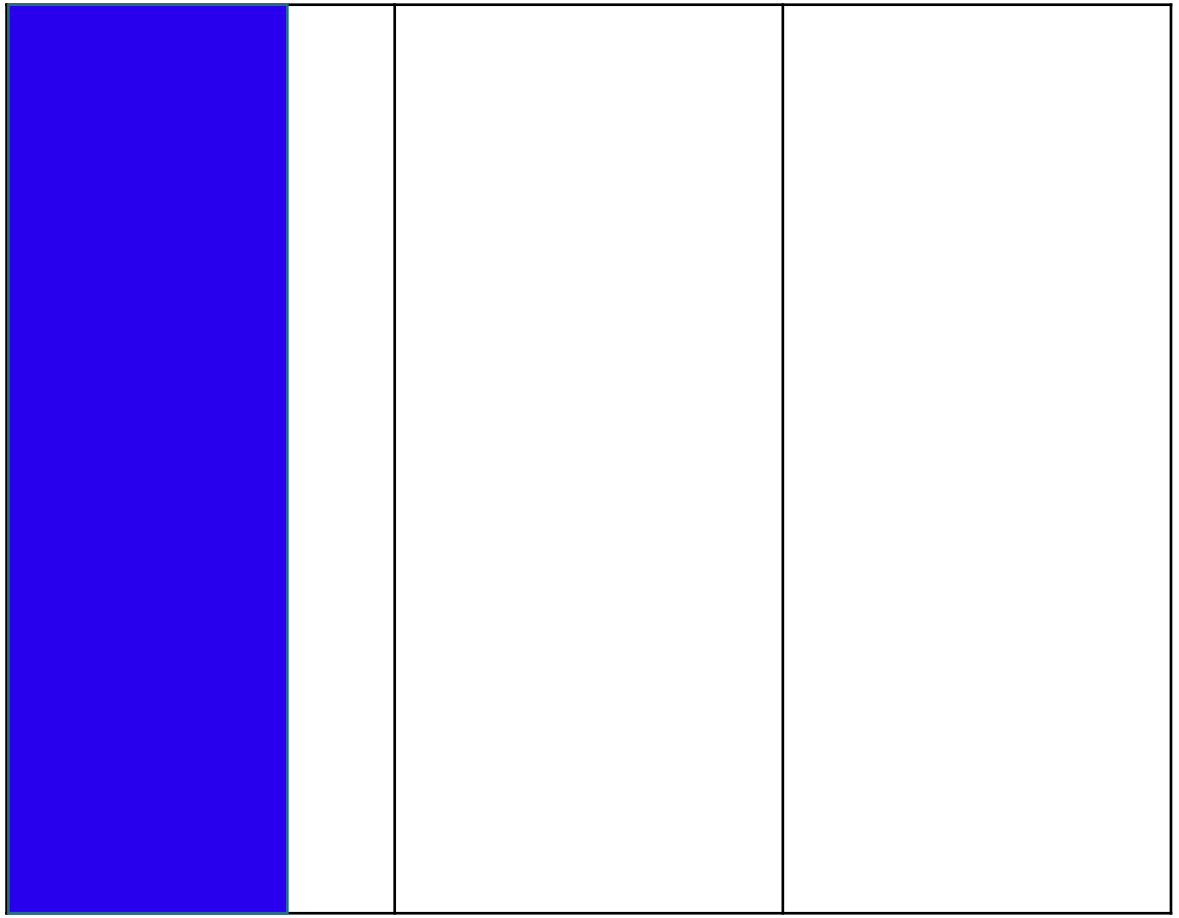


BreakToken

consumed_block_size=100
child_break_tokens

Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}

#child {
  height: 250px;
  background: blue;
}
```

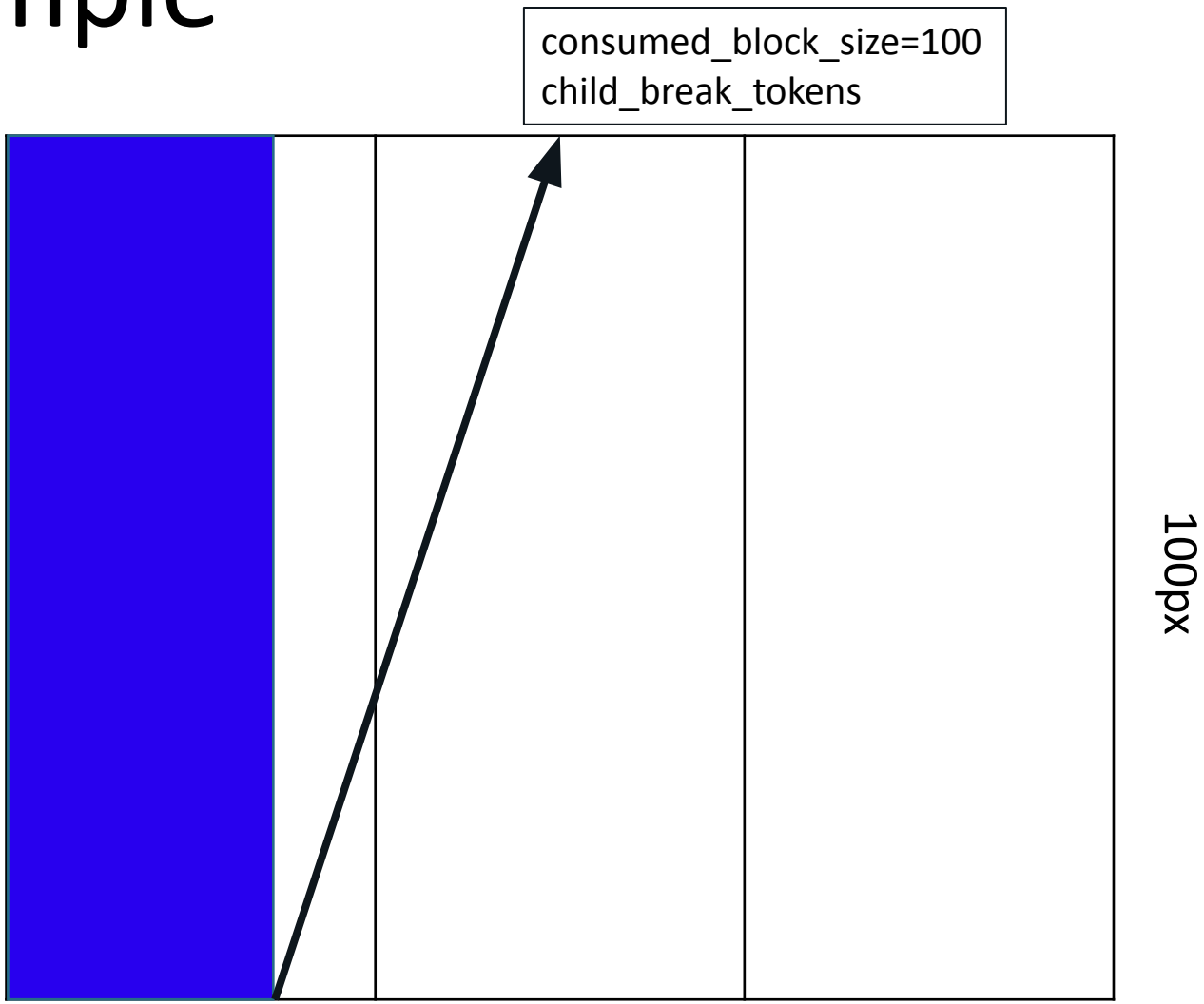


BreakToken

consumed_block_size=100
child_break_tokens

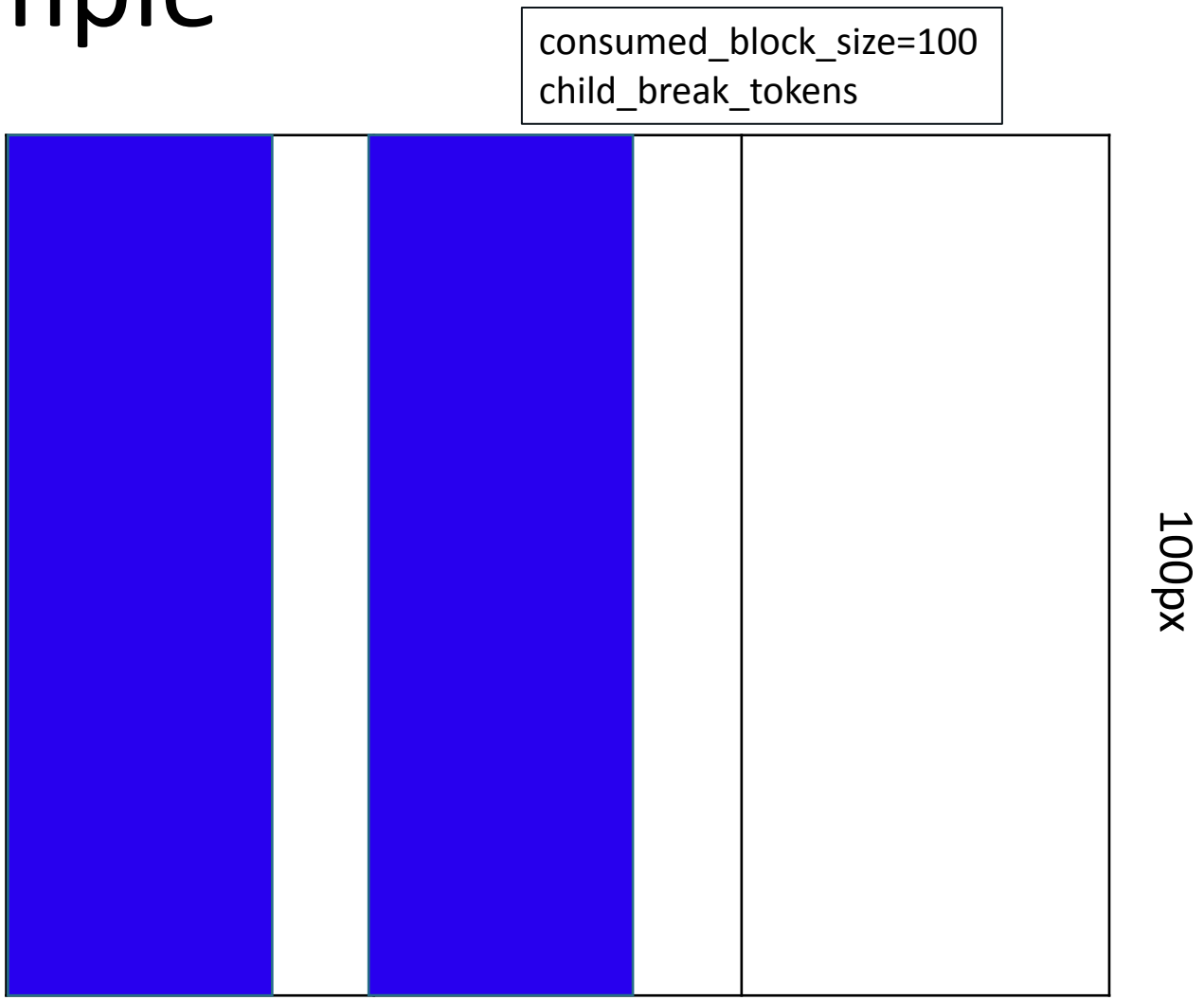
Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}

#child {
  height: 250px;
  background: blue;
}
```



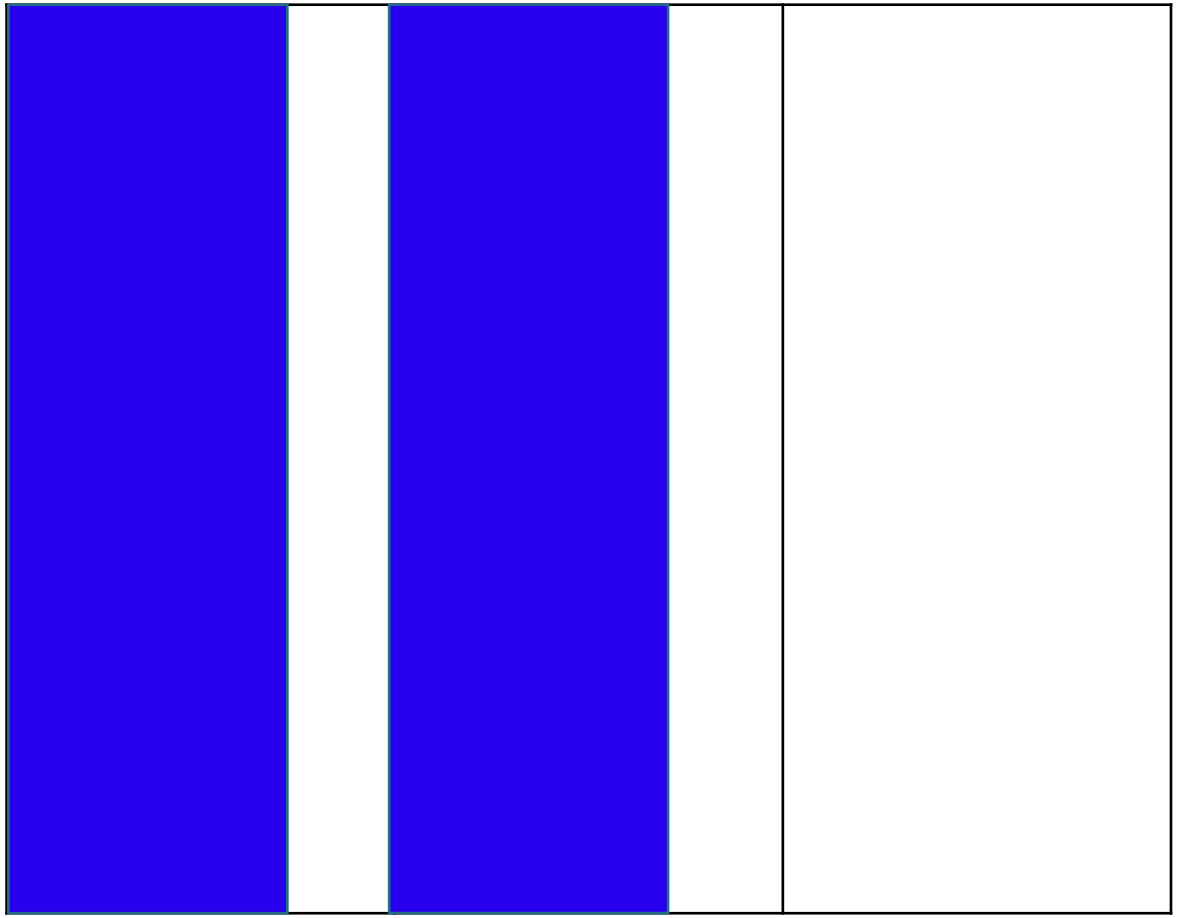
Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}

#child {
  height: 250px;
  background: blue;
}
```



Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}

#child {
  height: 250px;
  background: blue;
}
```

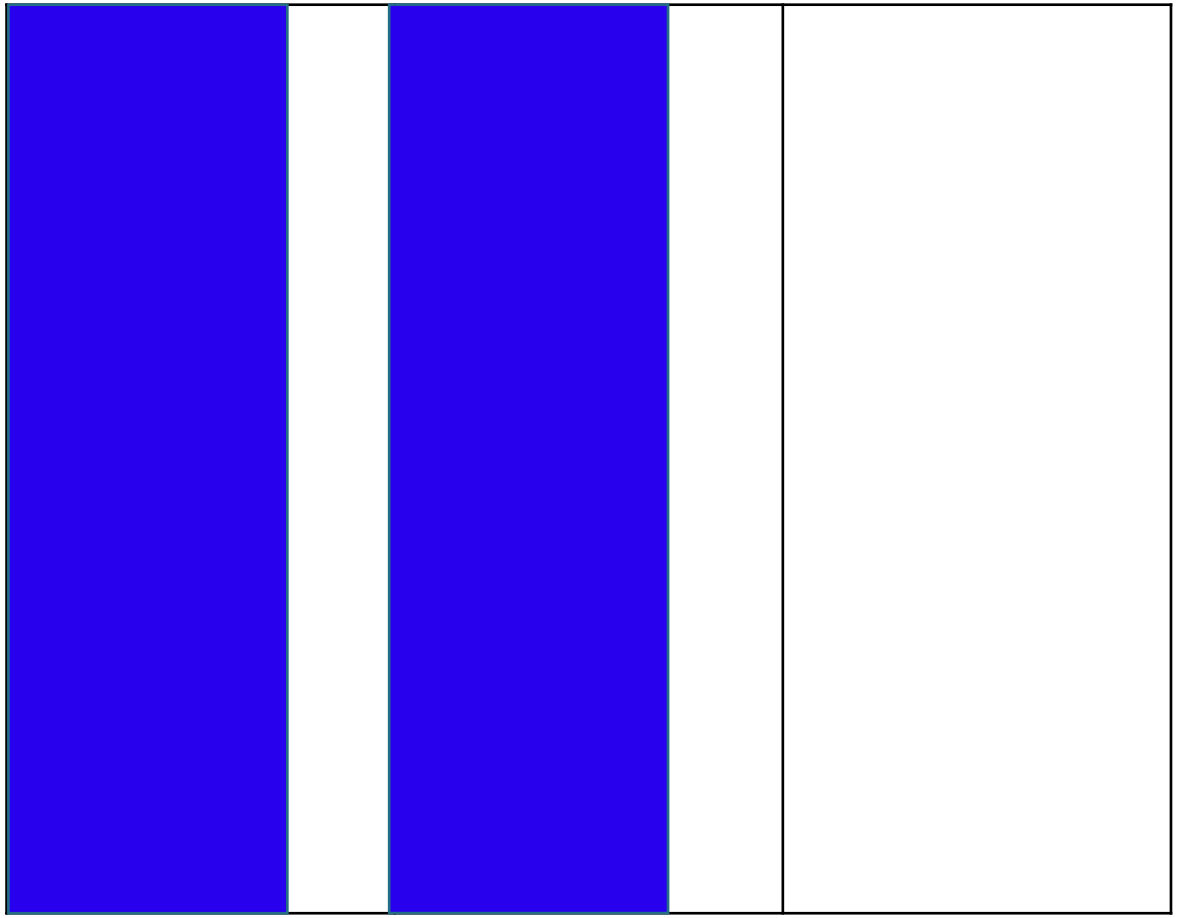


BreakToken

consumed_block_size
child_break_tokens

Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}

#child {
  height: 250px;
  background: blue;
}
```

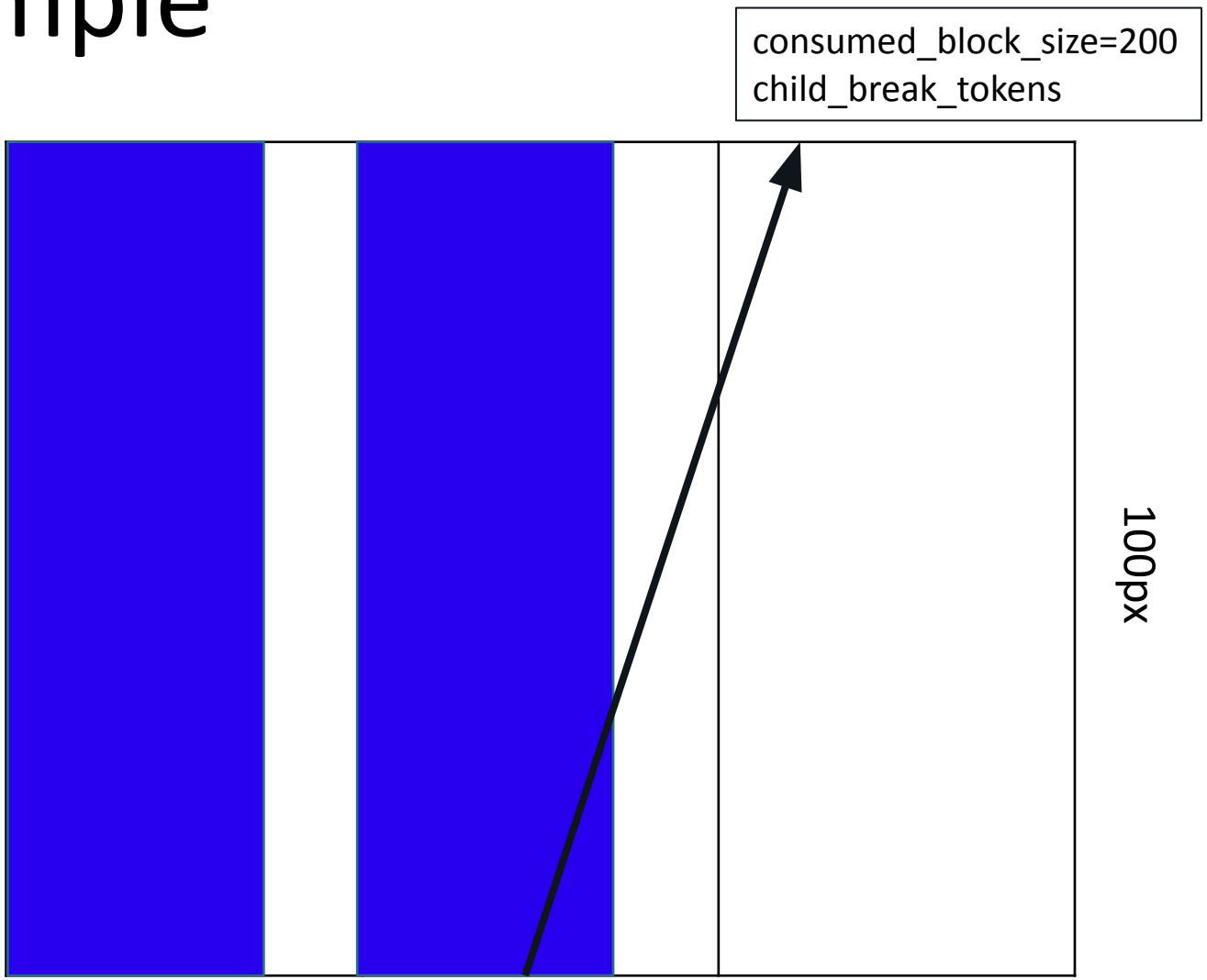


BreakToken

consumed_block_size=200
child_break_tokens

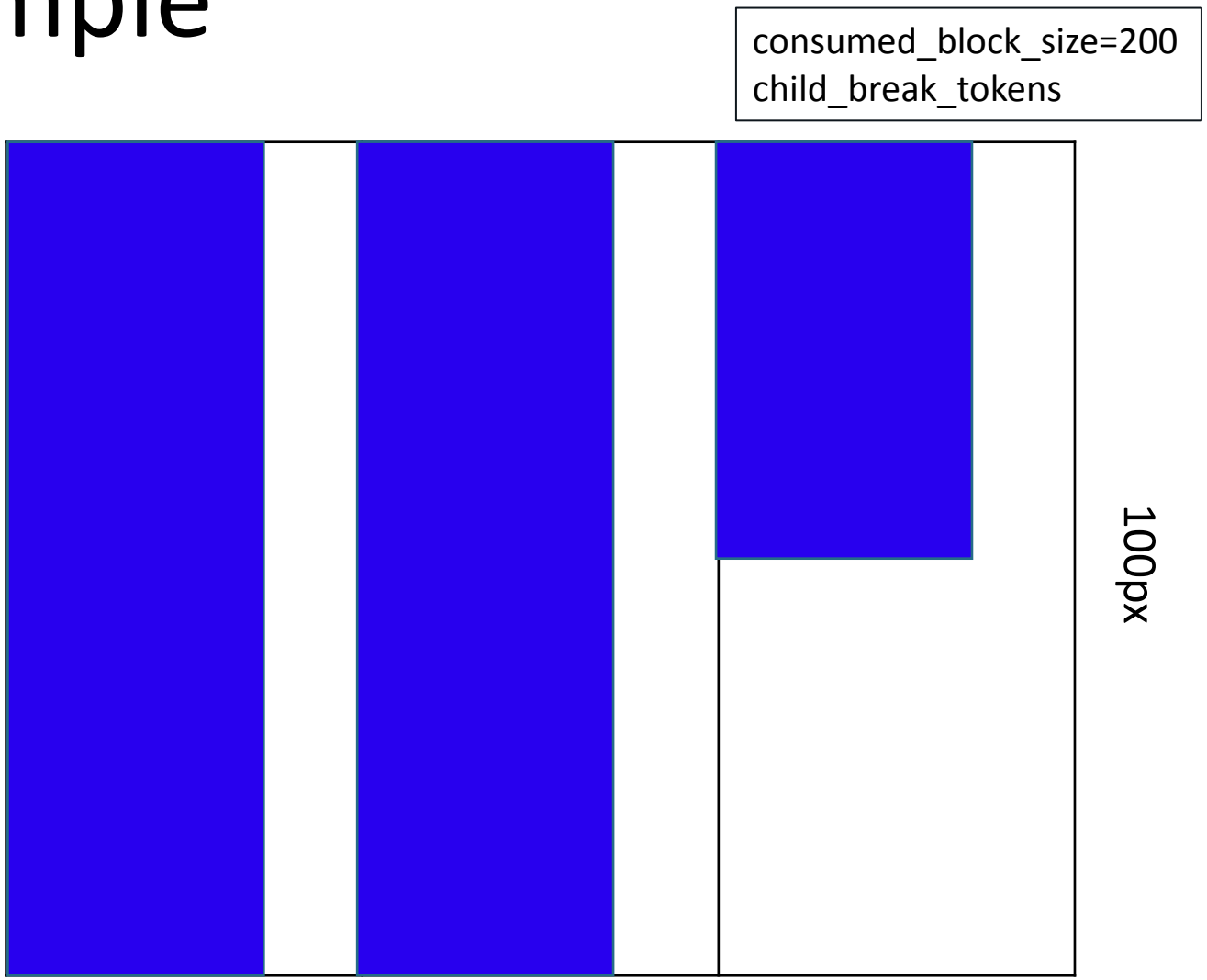
Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}

#child {
  height: 250px;
  background: blue;
}
```



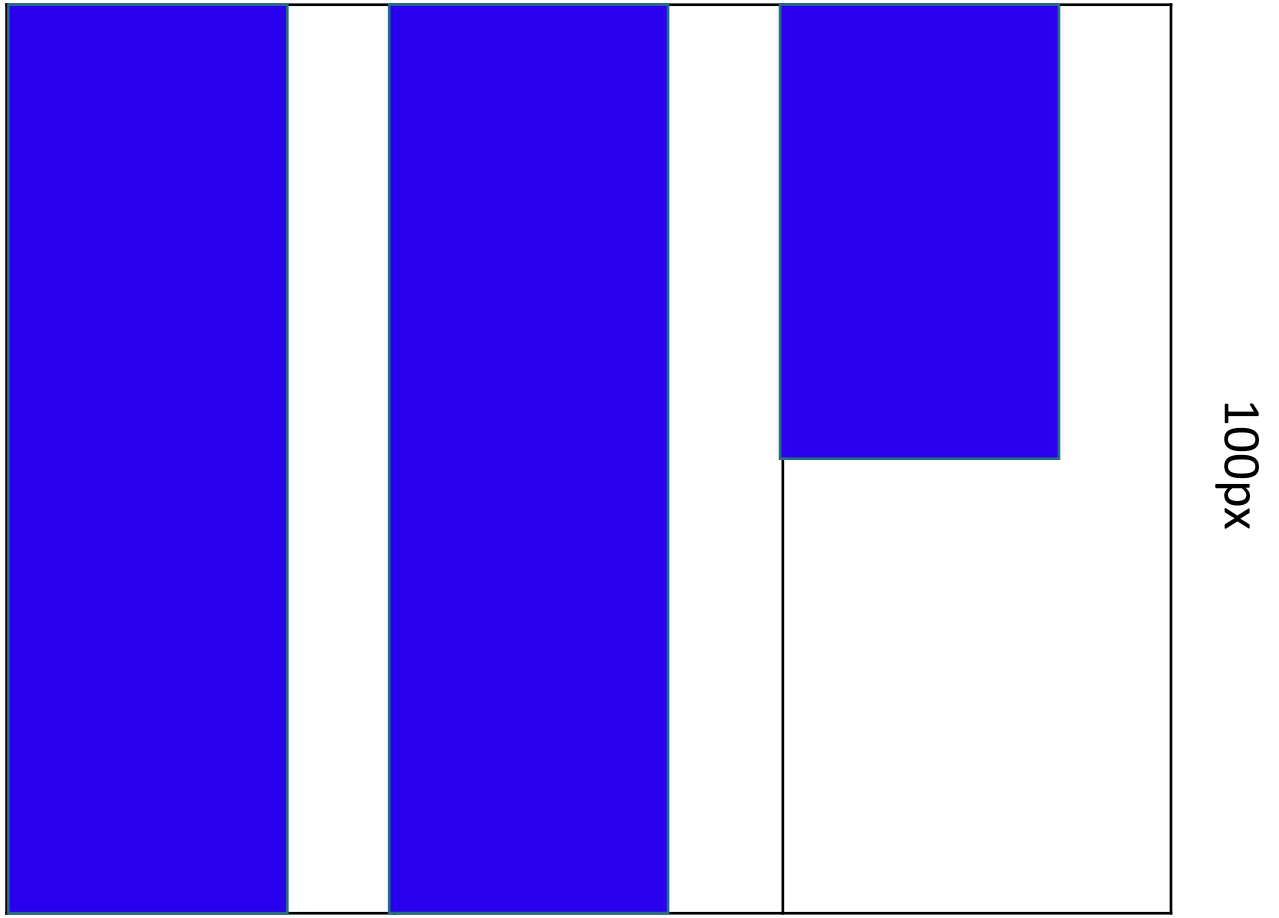
Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}

#child {
  height: 250px;
  background: blue;
}
```



Multi-column Example

```
.multicol {
  column-count: 3;
  height: 100px;
}

#child {
  height: 250px;
  background: blue;
}
```



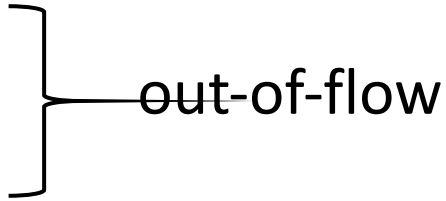
Out-of-flow Positioned Overview

CSS position property

- static
- relative
- fixed
- absolute
- sticky

CSS position property

- static
- relative
- fixed
- absolute
- sticky



CSS position property

- static
- relative
- fixed
- **absolute**
- sticky

out-of-flow

‘position: absolute’

- Taken out of flow
- Positioned relative its containing block
  - i.e. Relative to its nearest positioned ancestor

‘position: absolute’

- Taken out of flow
- Positioned relative its containing block
  - i.e. Relative to its nearest positioned ancestor

```
<div style=“position: relative;”>
  <div>
    <div style=“position: absolute; bottom:0;”></div>
  </div>
</div>
```

‘position: absolute’

- Taken out of flow
- Positioned relative its containing block
  - i.e. Relative to its nearest positioned ancestor

```
<div style=“position: relative;”>
  <div>
    <div style=“position: absolute; bottom:0;”></div>
  </div>
</div>
```

Out-of-flow Positioned in LayoutNG

Abspos Layout

```
<div style=“position: relative;”>
    <div>
        <div style=“position: absolute; bottom:0;”></div>
    </div>
</div>
```

Abspos Layout

```
<div style="position: relative;">
    <div>
        <div style="position: absolute; bottom:0;"></div>
    </div>
</div>
```

Abspos Layout

```
<div style="position: relative;">
    <div>
        <div style="position: absolute; bottom:0;"></div>
    </div>
</div>
```

Abspos Layout

```
<div style="position: relative;">
    <div>
        <div style="position: absolute; bottom:0;"></div>
    </div>
</div>
```

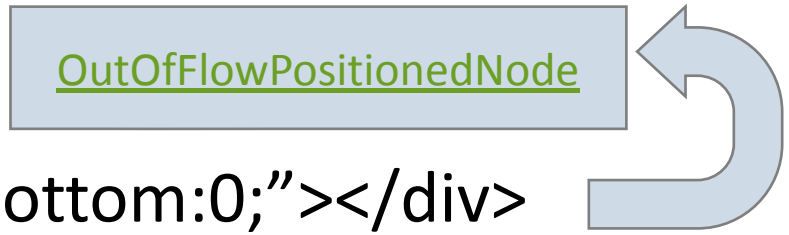
Abspos Layout

```
<div style="position: relative;">
    <div>
        <div style="position: absolute; bottom:0;"></div>
    </div>
</div>
```

OutOfFlowPositionedNode

Abspos Layout

```
<div style="position: relative;">
    <div>
        <div style="position: absolute; bottom:0;"></div>
    </div>
</div>
```



Abspos Layout

** **


```
    <div>
        <div style="position: absolute; bottom:0;"></div>
    </div>
```

** **

Abspos Layout

** ** OutOfFlowPositionedNode 

```
    <div>
        <div style="position: absolute; bottom:0;"></div>
    </div>
```

** **

OutOfFlowPositionedNode

- Block node
  - The out-of-flow node that we need to perform layout on
- Static position
  - The offset the out-of-flow node had it been in-flow

Out-of-flow Positioned Fragmentation

Issue #1

** ** OutOfFlowPositionedNode 

```
    <div>
        <div style="position: absolute; bottom:0;"></div>
    </div>
```

** **

Issue #1

```
<div style="columns: 3;">
    <div style="position: relative;">
        <div>
            <div style="position: absolute; bottom:0;"></div>
        </div>
    </div>
</div>
```

OutOfFlowPositionedNode

Issue #1

```
<div style="columns: 3;">
    <div style="position: relative;">
        <div>
            <div style="position: absolute; bottom:0;"></div>
        </div>
    </div>
</div>
```

OutOfFlowPositionedNode

Issue #1

** **

```
    <div style="position: relative;">
        <div>
            <div style="position: absolute; bottom:0;"></div>
        </div>
    </div>
```

** **

OutOfFlowPositionedNode

Issue #1

** ** OutOfFlowPositionedNode 

```
    <div style="position: relative;">
        <div>
            <div style="position: absolute; bottom:0;"></div>
        </div>
    </div>
</div>
```

Out-of-flow Fragmentation algorithm

Step 1: Bubble nodes up to fragmentation context root

Issue #2

- We are performing layout at a later point in time
- We need more information to position correctly
  - What is the containing block?
  - Where is the containing block located?

Out-of-flow Fragmentation algorithm

Step 1: Bubble nodes up to fragmentation context root

- Collecting extra info in OutOfFlowPositionedNode

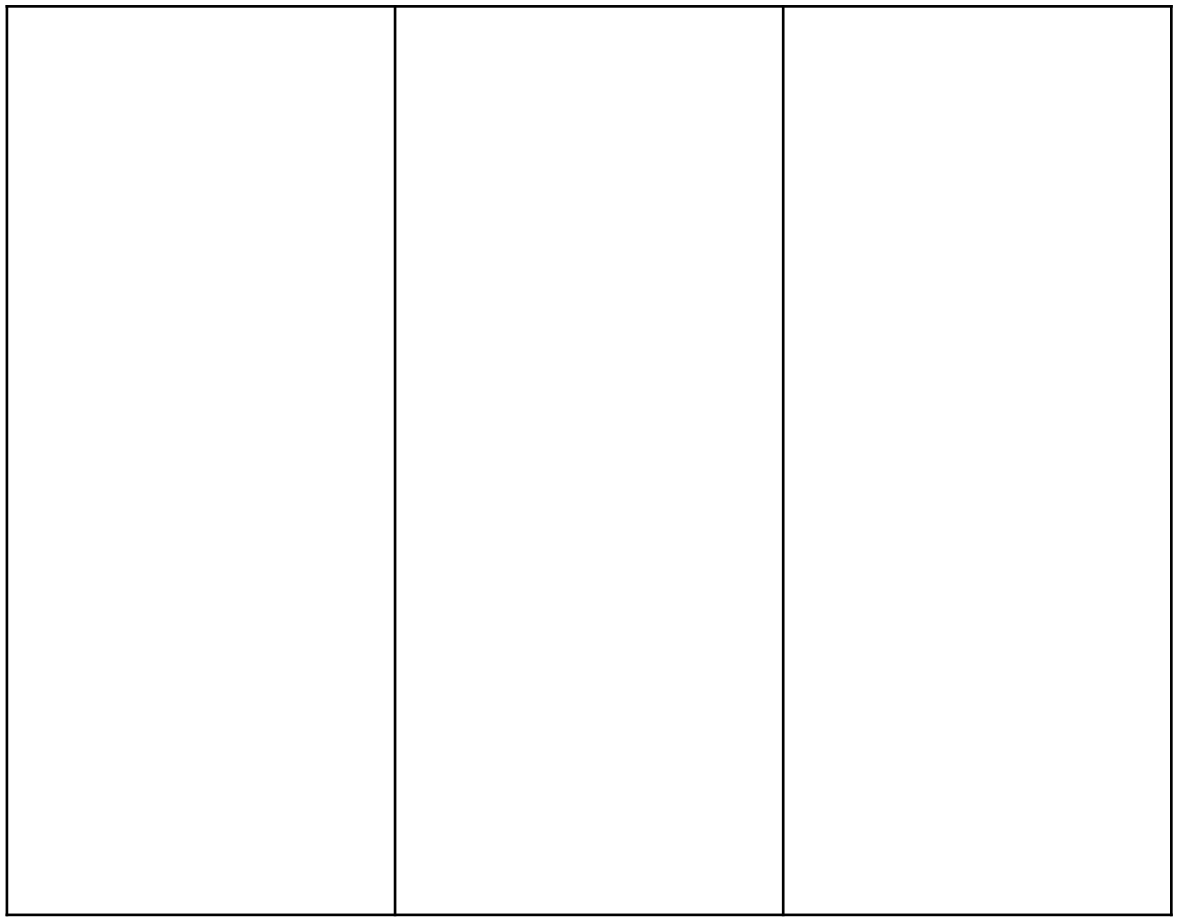
OutOfFlowPositionedNode

- Block node
  - The out-of-flow node that we need to perform layout on
- Static position
  - The offset the out-of-flow node had it been in-flow

OutOfFlowPositionedNode

- Block node
  - The out-of-flow node that we need to perform layout on
- Static position
  - The offset the out-of-flow node had it been in-flow
- Containing block physical fragment
  - Reference to the containing block we are positioned relative to
- Containing block offset
  - The offset from the first containing block fragment to the fragmentation context root (as if all fragmentainers were lined up in the block direction)

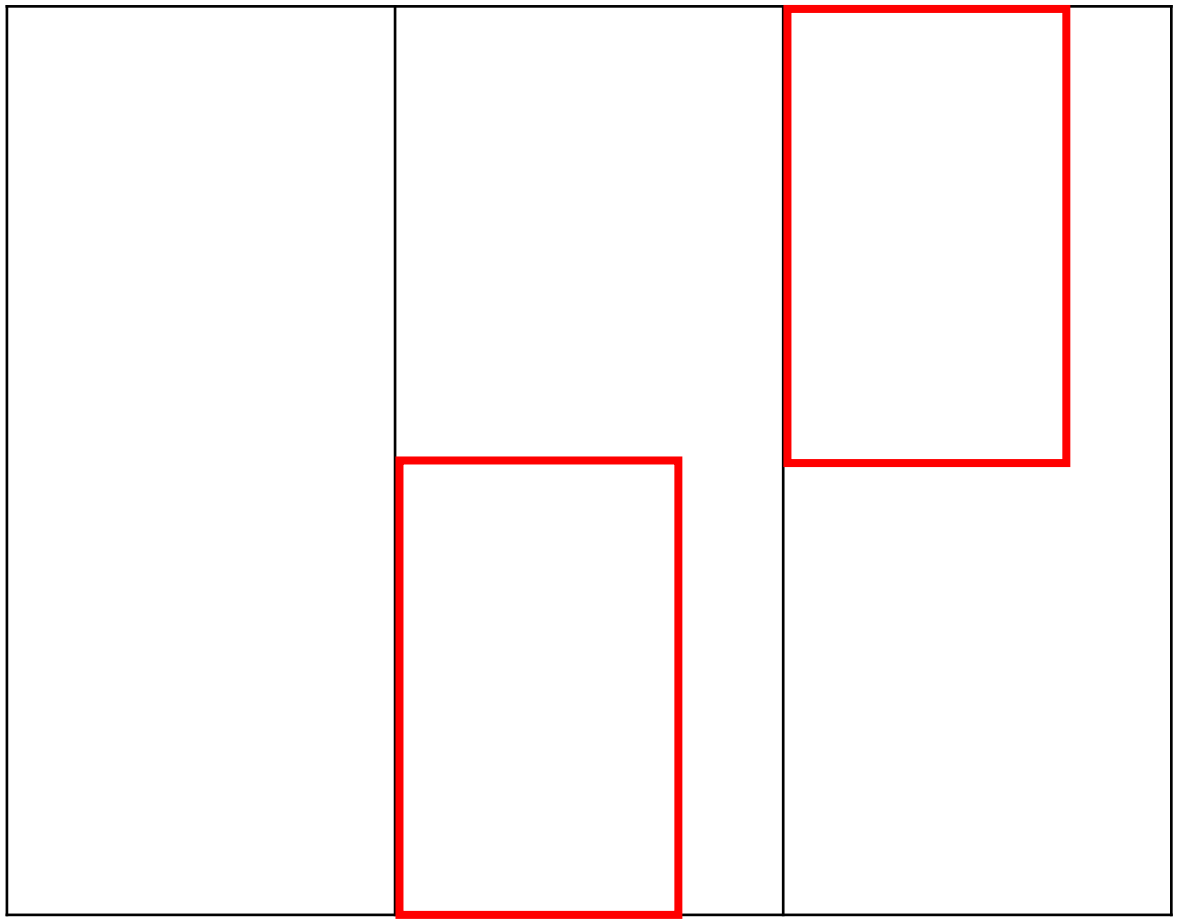
OutOfFlowPositionedNode



OutOfFlowPositionedNode

- Block Node
- Static position
- Containing block physical fragment
- Containing block offset

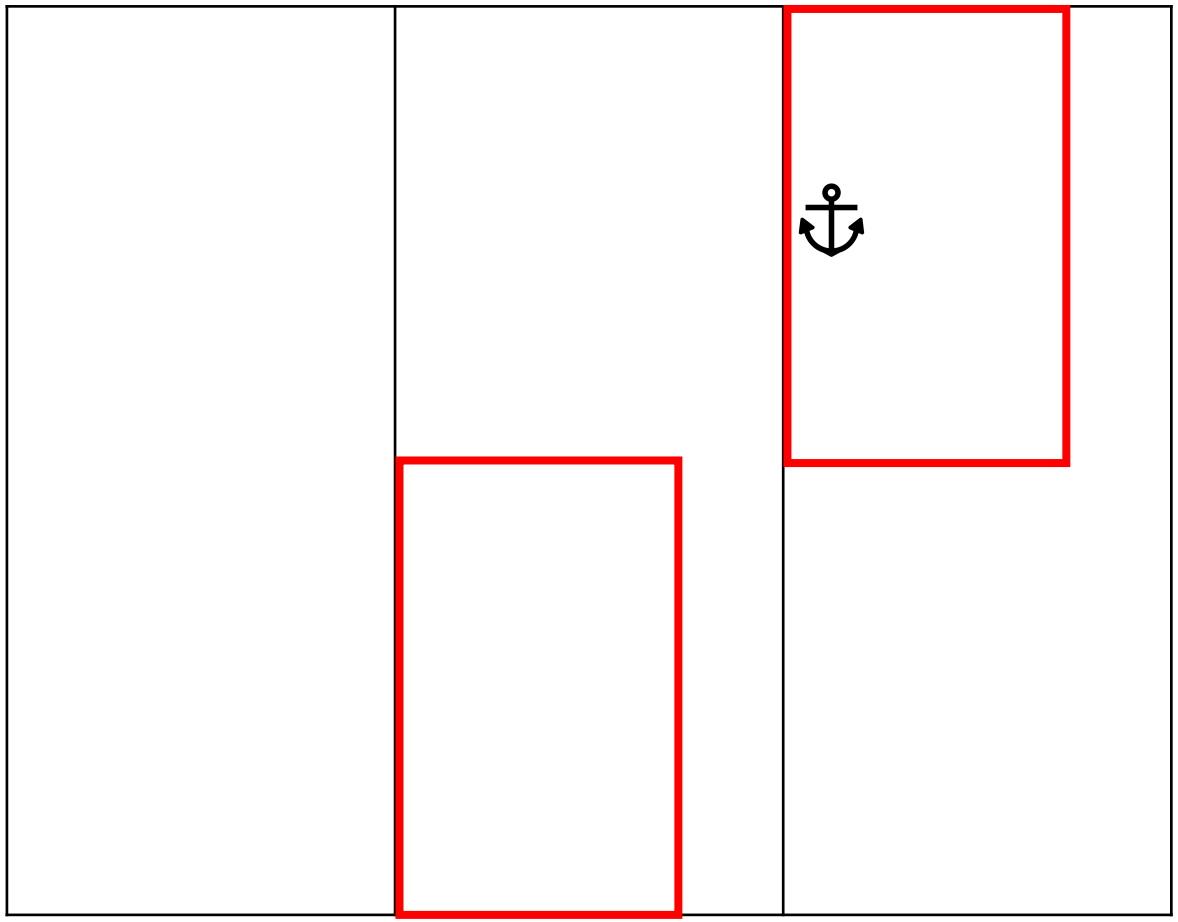
OutOfFlowPositionedNode



OutOfFlowPositionedNode

- Block Node
- Static position
- Containing block physical fragment
- Containing block offset

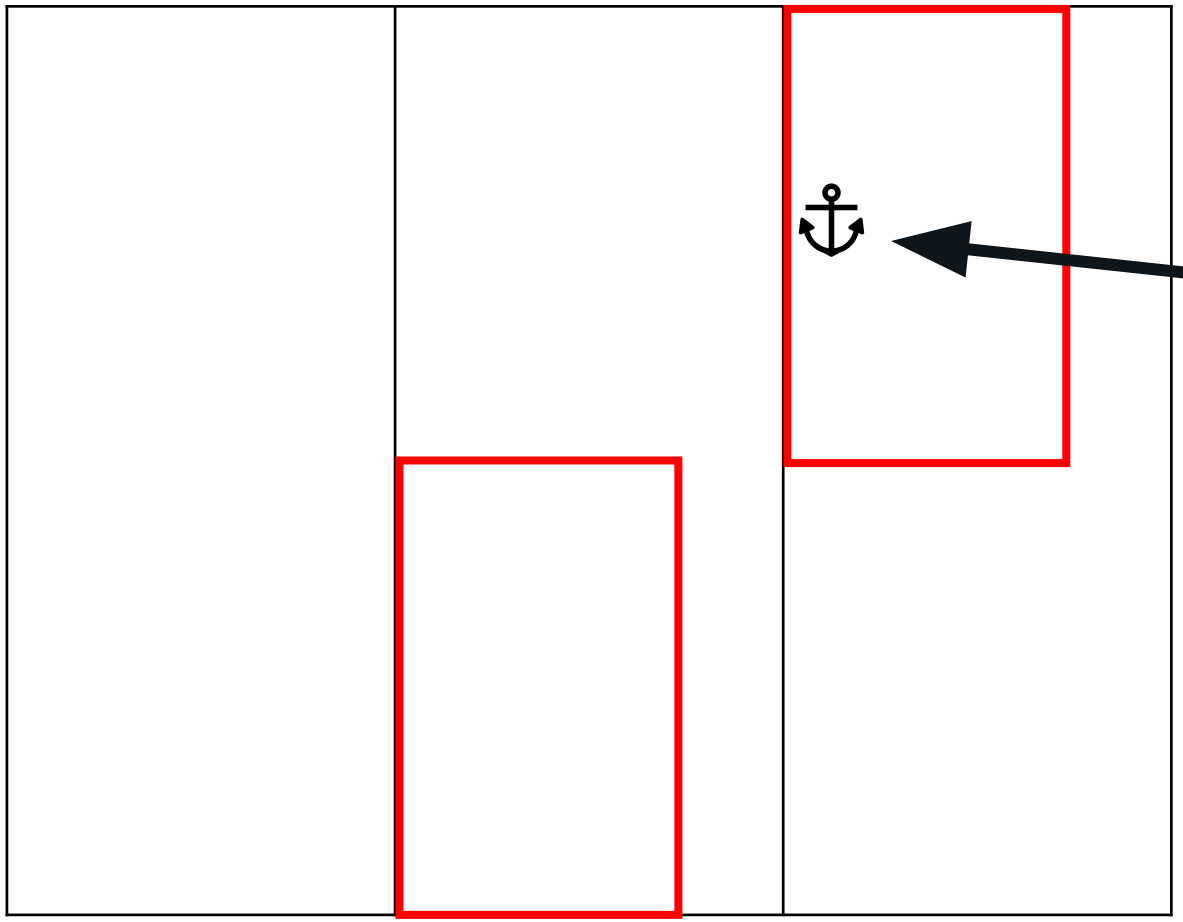
OutOfFlowPositionedNode



OutOfFlowPositionedNode

- Block Node
- Static position
- Containing block physical fragment
- Containing block offset

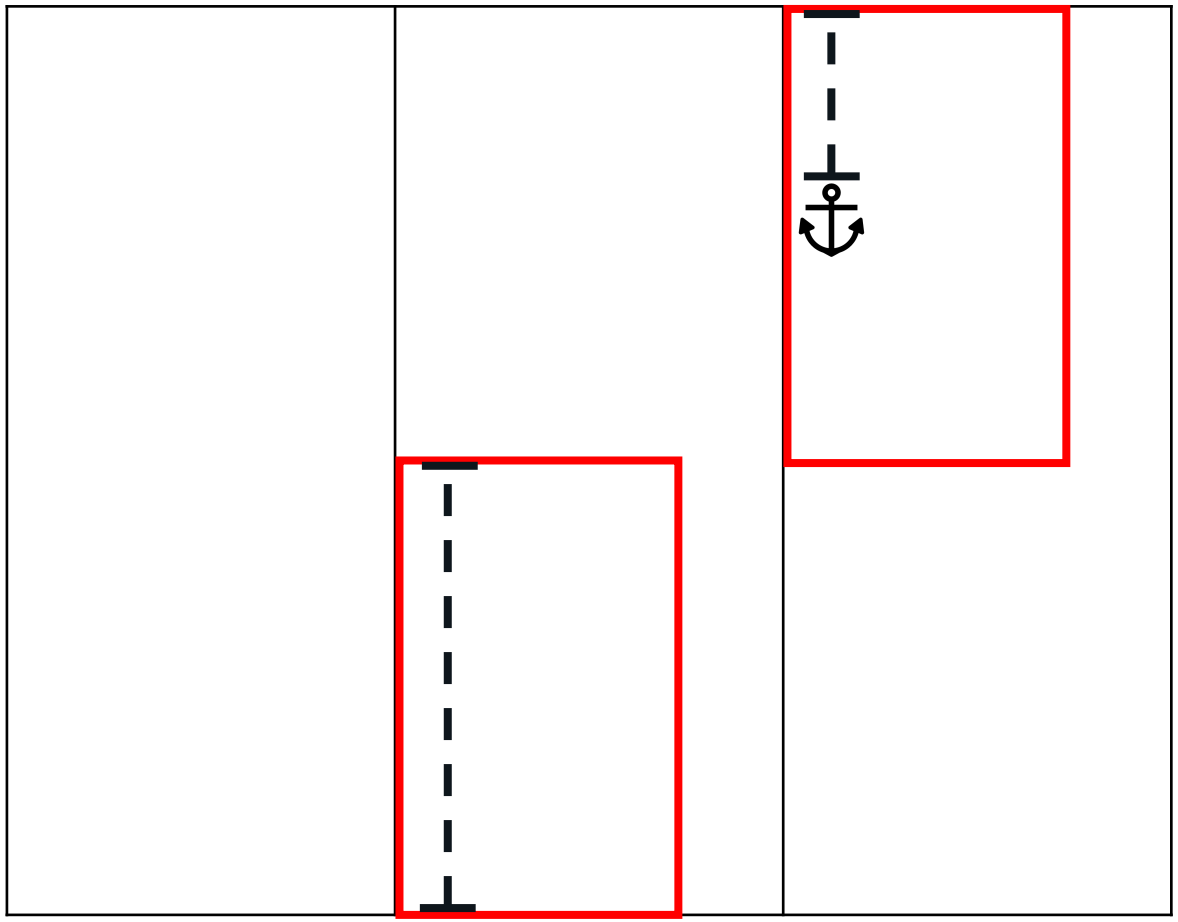
OutOfFlowPositionedNode



OutOfFlowPositionedNode

- **Block Node**
- Static position
- Containing block physical fragment
- Containing block offset

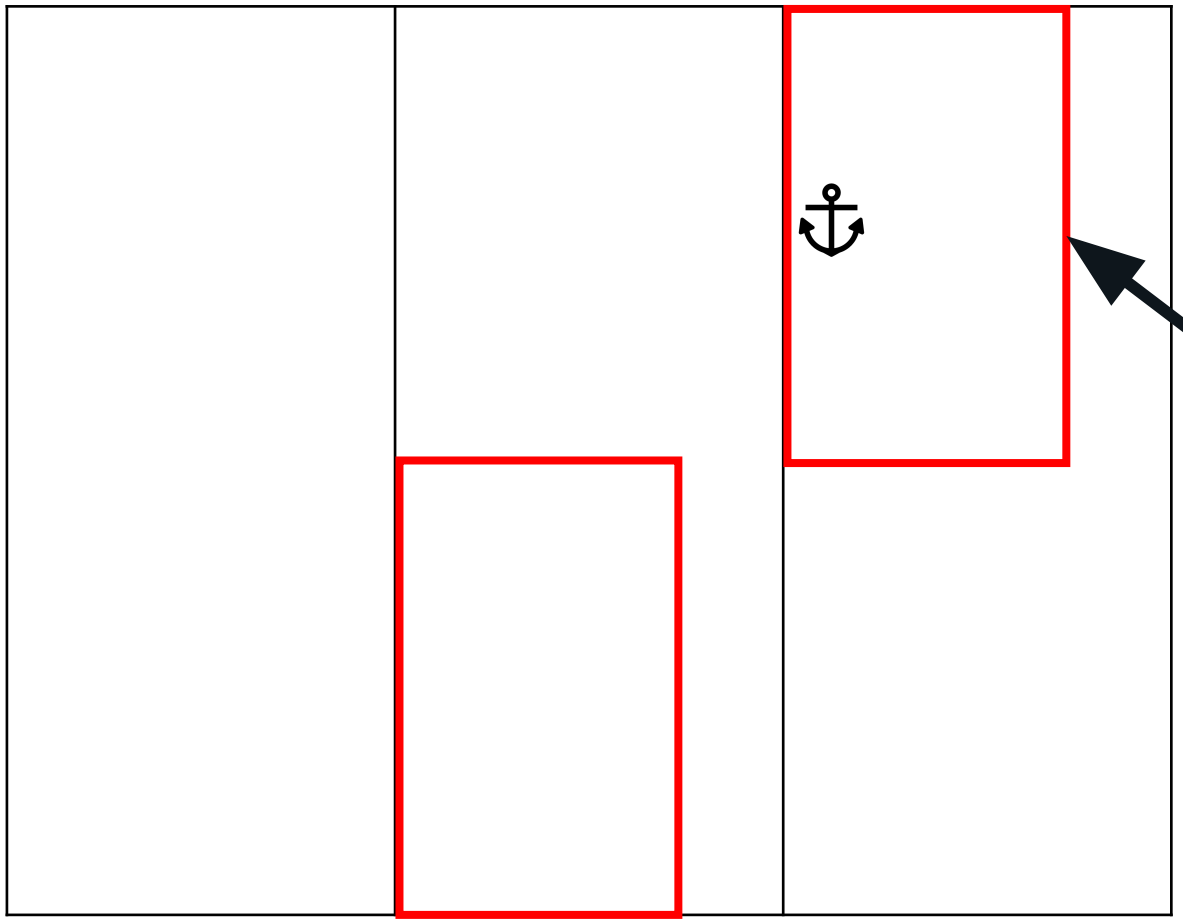
OutOfFlowPositionedNode



OutOfFlowPositionedNode

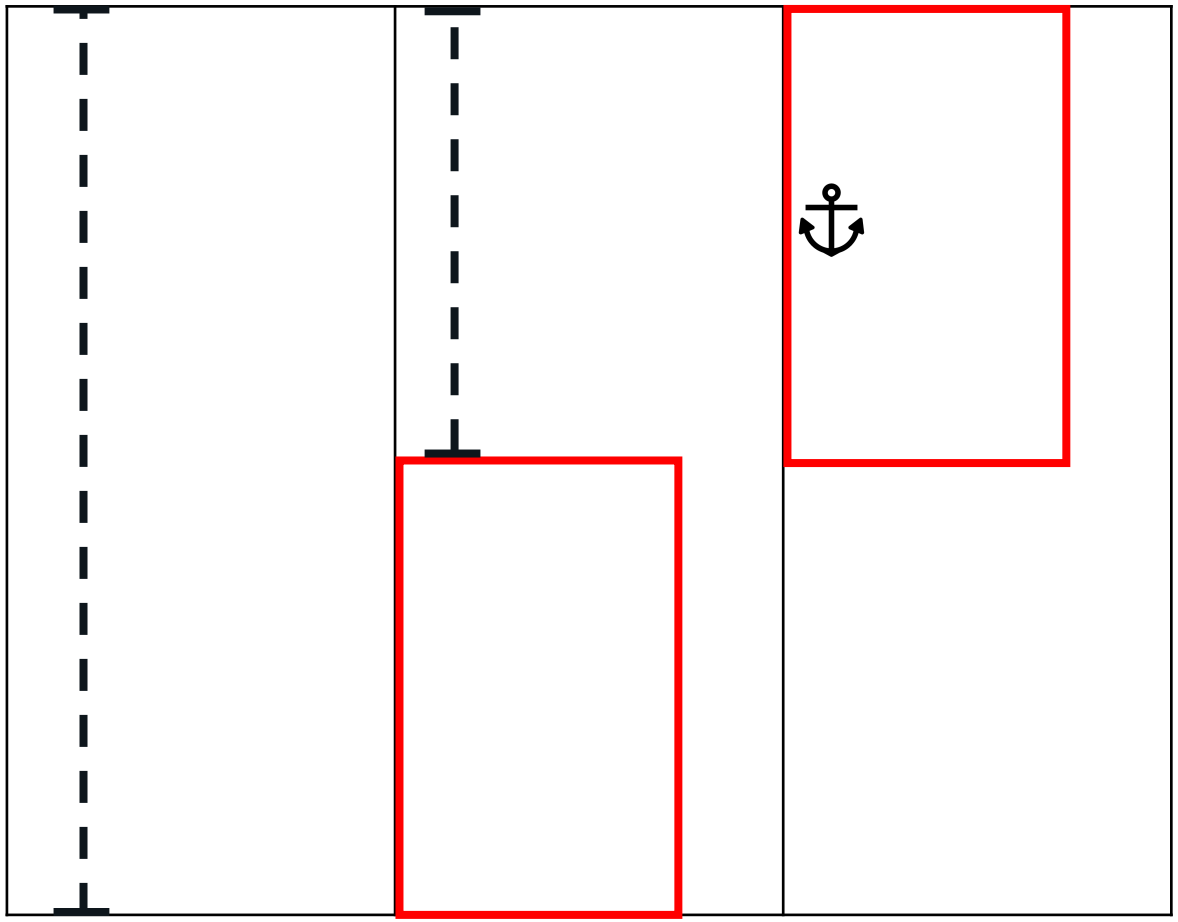
- Block Node
- **Static position**
- Containing block physical fragment
- Containing block offset

OutOfFlowPositionedNode



OutOfFlowPositionedNode
- Block Node
- Static position
- **Containing block physical fragment**
- Containing block offset

OutOfFlowPositionedNode



OutOfFlowPositionedNode

- Block Node
- Static position
- Containing block physical fragment
- **Containing block offset**

Out-of-flow Fragmentation algorithm

Step 1: Bubble nodes up to fragmentation context root

- Collecting extra info in OutOfFlowPositionedNode

Out-of-flow Fragmentation algorithm

Step 1: Bubble nodes up to fragmentation context root

- Collecting extra info in OutOfFlowPositionedNode

Step 2: Layout the OOF positioned nodes

Out-of-flow Fragmentation algorithm

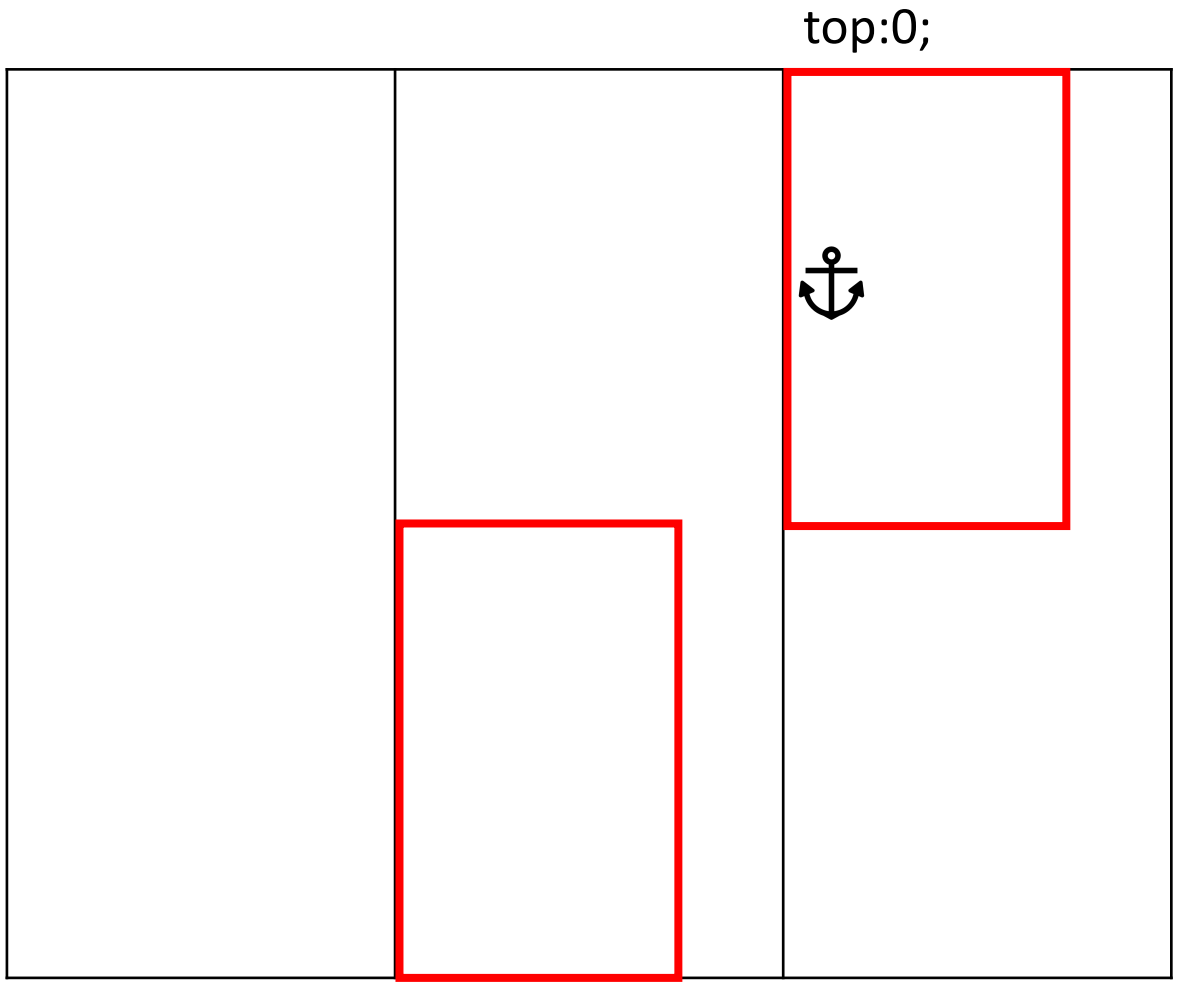
Step 1: Bubble nodes up to fragmentation context root

- Collecting extra info in OutOfFlowPositionedNode

Step 2: Layout the OOF positioned nodes

- Start offset

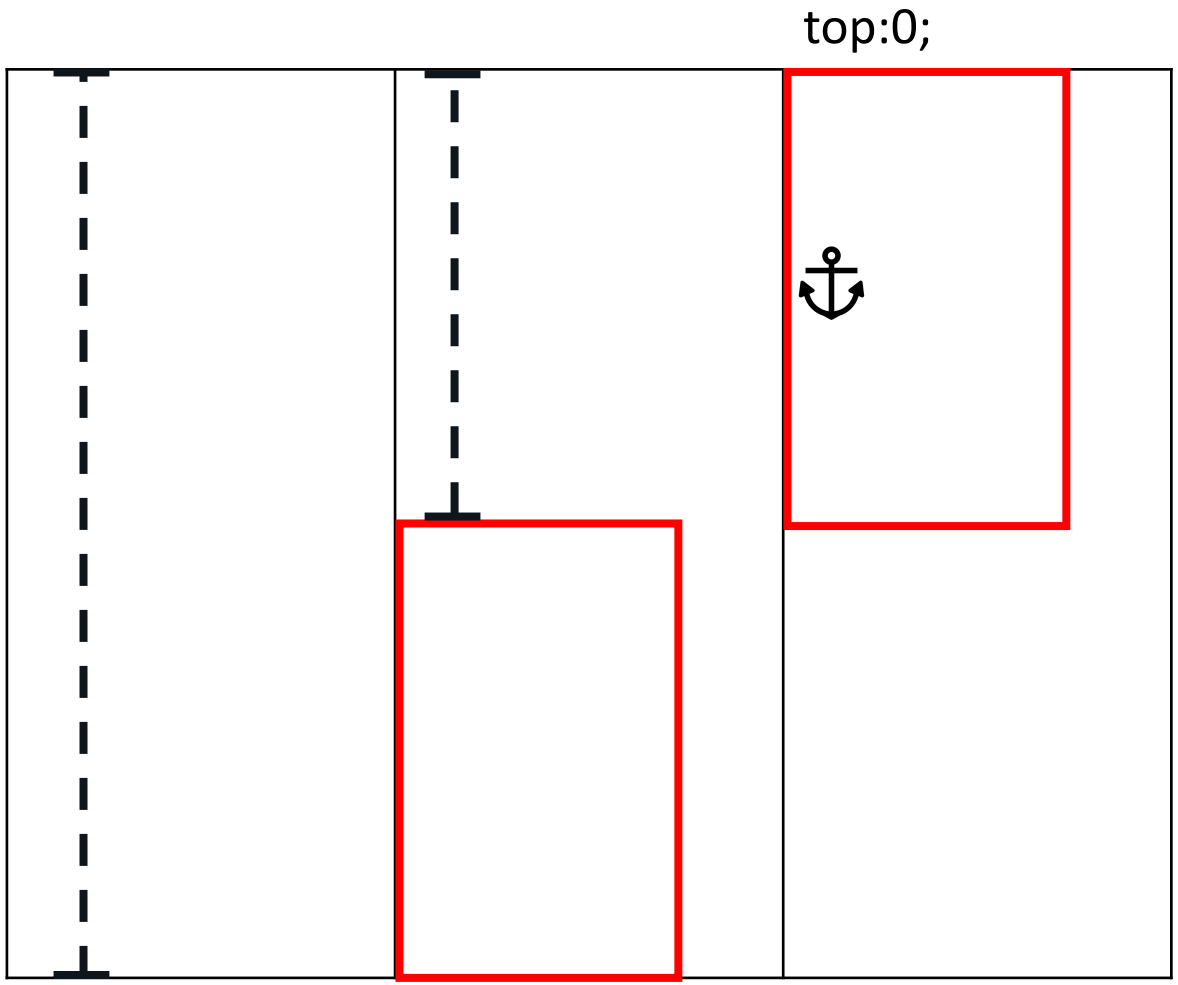
Start Offset



OutOfFlowPositionedNode

- Block Node
- Static position
- Containing block physical fragment
- Containing block offset

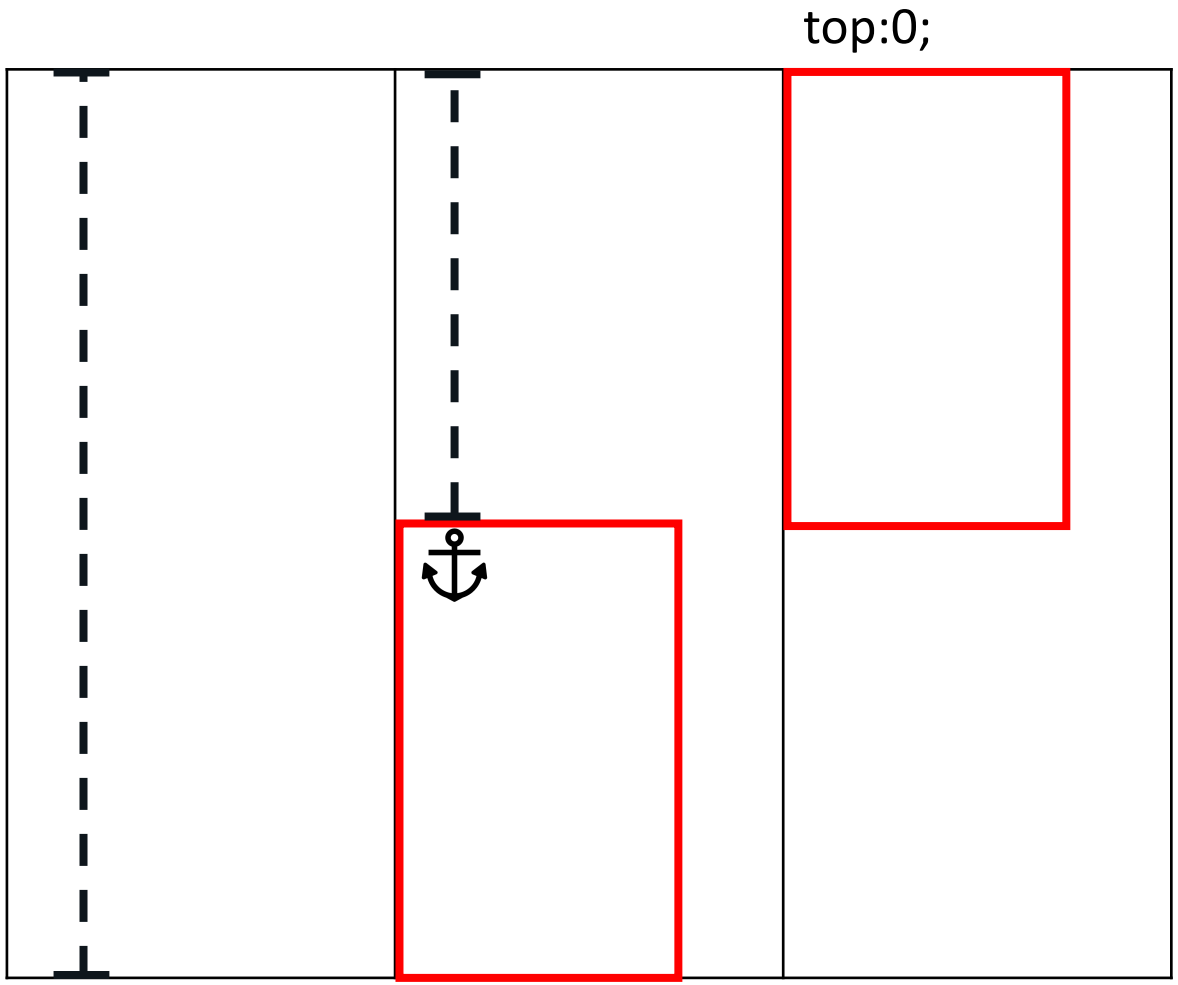
Start Offset



OutOfFlowPositionedNode

- Block Node
- Static position
- Containing block physical fragment
- **Containing block offset**

Start Offset



OutOfFlowPositionedNode

- Block Node
- Static position
- Containing block physical fragment
- **Containing block offset**

Out-of-flow Fragmentation algorithm

Step 1: Bubble nodes up to fragmentation context root

- Collecting extra info in OutOfFlowPositionedNode

Step 2: Layout the OOF positioned nodes

- Start offset

Out-of-flow Fragmentation algorithm

Step 1: Bubble nodes up to fragmentation context root

- Collecting extra info in OutOfFlowPositionedNode

Step 2: Layout the OOF positioned nodes

- Start offset
- Fragmentainer to start layout

Out-of-flow Fragmentation algorithm

Step 1: Bubble nodes up to fragmentation context root

- Collecting extra info in OutOfFlowPositionedNode

Step 2: Layout the OOF positioned nodes

- Start offset
- Fragmentainer to start layout
- Constraint space

Out-of-flow Fragmentation algorithm

Step 1: Bubble nodes up to fragmentation context root

- Collecting extra info in OutOfFlowPositionedNode

Step 2: Layout the OOF positioned nodes

- Start offset
- Fragmentainer to start layout
- Constraint space
- Layout() -> LayoutResult

Out-of-flow Fragmentation algorithm

Step 1: Bubble nodes up to fragmentation context root

- Collecting extra info in OutOfFlowPositionedNode

Step 2: Layout the OOF positioned nodes

- Start offset
- Fragmentainer to start layout
- Constraint space
- Layout() -> LayoutResult
- Handle break tokens

Handle Break Tokens

- If the LayoutResult produced BreakToken
  - Determine the next fragmentainer
  - Create a new ConstraintSpace
  - Perform Layout
  - Repeat until no BreakToken

Out-of-flow Fragmentation algorithm

Step 1: Bubble nodes up to fragmentation context root

- Collecting extra info in OutOfFlowPositionedNode

Step 2: Layout the OOF positioned nodes

- Start offset
- Fragmentainer to start layout
- Constraint space
- Layout() -> LayoutResult
- Handle break tokens

Step 3: Transplant the OOF fragments into fragmentainers

Issue #3

- We want to add the OOF fragments to fragmentainers
- Fragmentainers have already finished layout
- PhysicalFragments are immutable

Out-of-flow Fragmentation algorithm

Step 1: Bubble nodes up to fragmentation context root

- Collecting extra info in OutOfFlowPositionedNode

Step 2: Layout the OOF positioned nodes

- Start offset
- Fragmentainer to start layout
- Constraint space
- Layout() -> LayoutResult
- Handle break tokens

Step 3: Transplant the OOF fragments into fragmentainers

- Clone fragmentainer

Out-of-flow Fragmentation algorithm

Step 1: Bubble nodes up to fragmentation context root

- Collecting extra info in OutOfFlowPositionedNode

Step 2: Layout the OOF positioned nodes

- Start offset
- Fragmentainer to start layout
- Constraint space
- Layout() -> LayoutResult
- Handle break tokens

Step 3: Transplant the OOF fragments into fragmentainers

- Clone fragmentainer
- Add OOF children

Out-of-flow Fragmentation algorithm

Step 1: Bubble nodes up to fragmentation context root

- Collecting extra info in OutOfFlowPositionedNode

Step 2: Layout the OOF positioned nodes

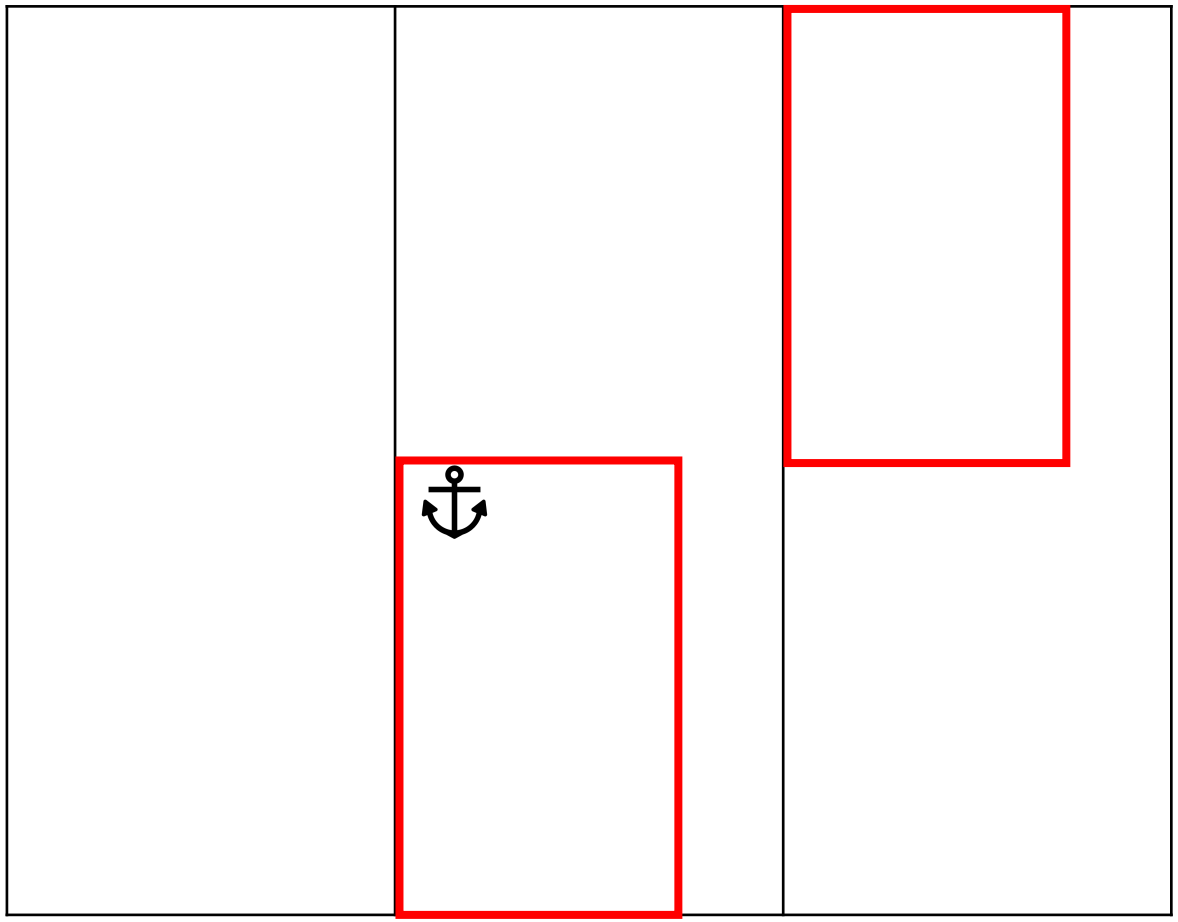
- Start offset
- Fragmentainer to start layout
- Constraint space
- Layout() -> LayoutResult
- Handle break tokens

Step 3: Transplant the OOF fragments into fragmentainers

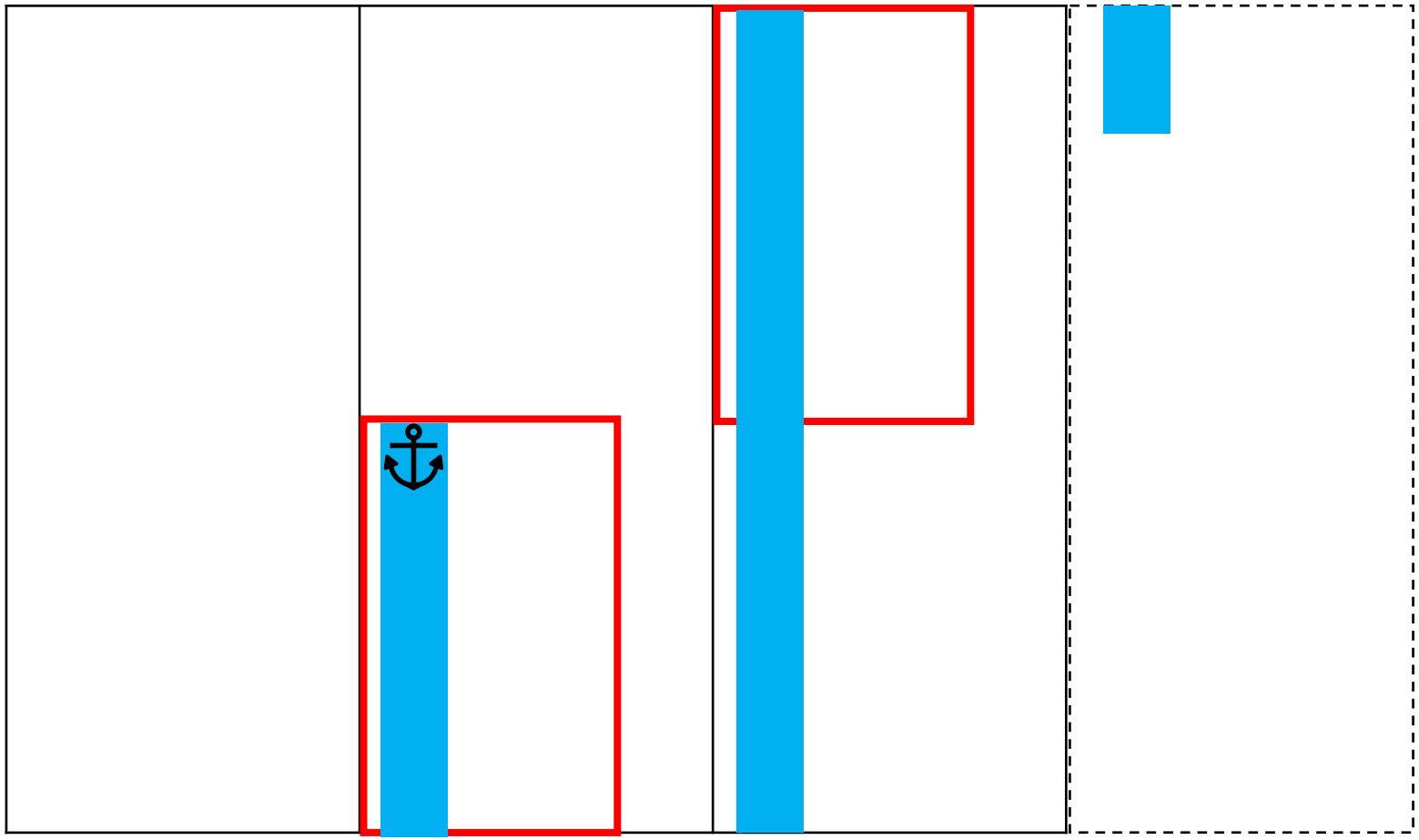
- Clone fragmentainer
- Add OOF children

Step 4: Add new fragmentainers

Add New Fragmentainers



Add New Fragmentainers



Out-of-flow Fragmentation algorithm ☑

Step 1: Bubble nodes up to fragmentation context root

- Collecting extra info in OutOfFlowPositionedNode

Step 2: Layout the OOF positioned nodes

- Start offset
- Fragmentainer to start layout
- Constraint space
- Layout() -> LayoutResult
- Handle break tokens

Step 3: Transplant the OOF fragments into fragmentainers

- Clone fragmentainer
- Add OOF children

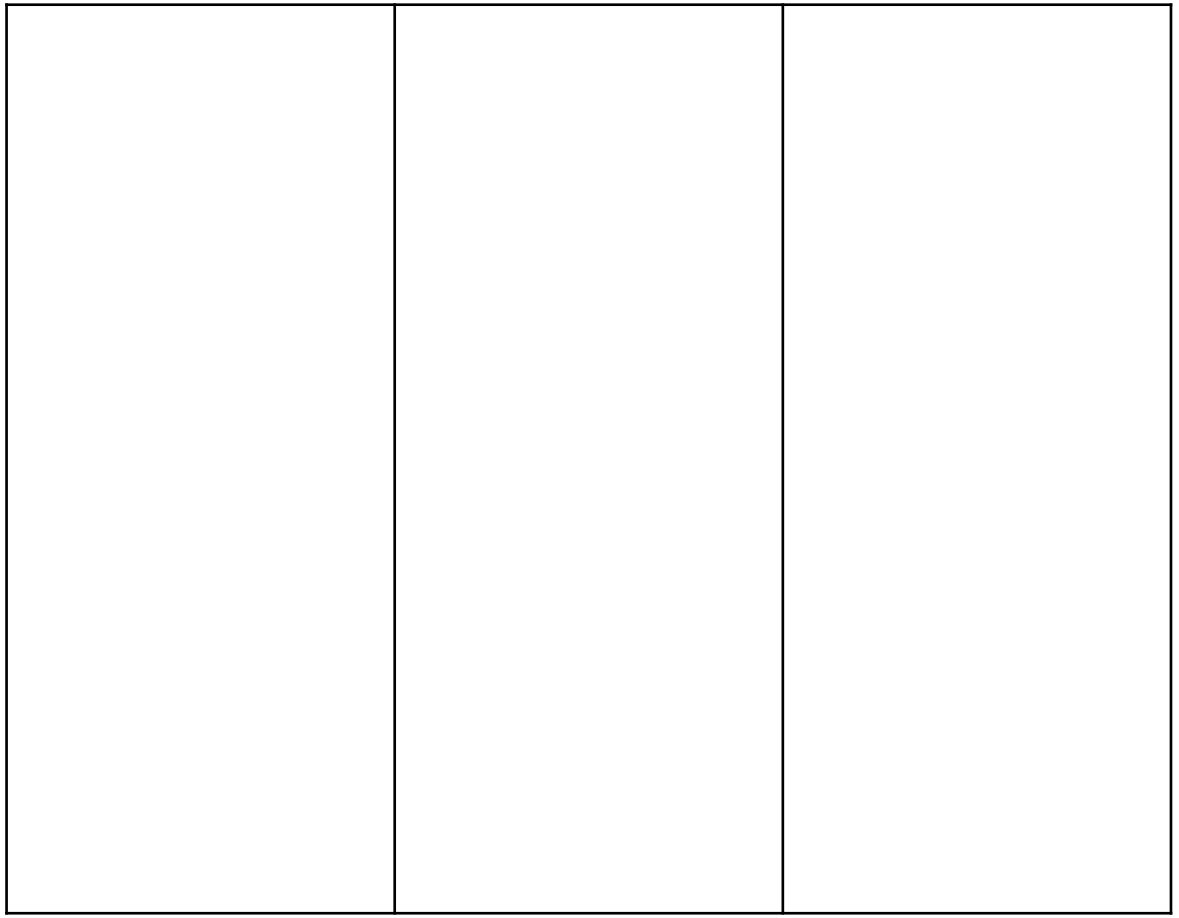
Step 4: Add new fragmentainers

What’s Next?

Example: height:auto; bottom:0;

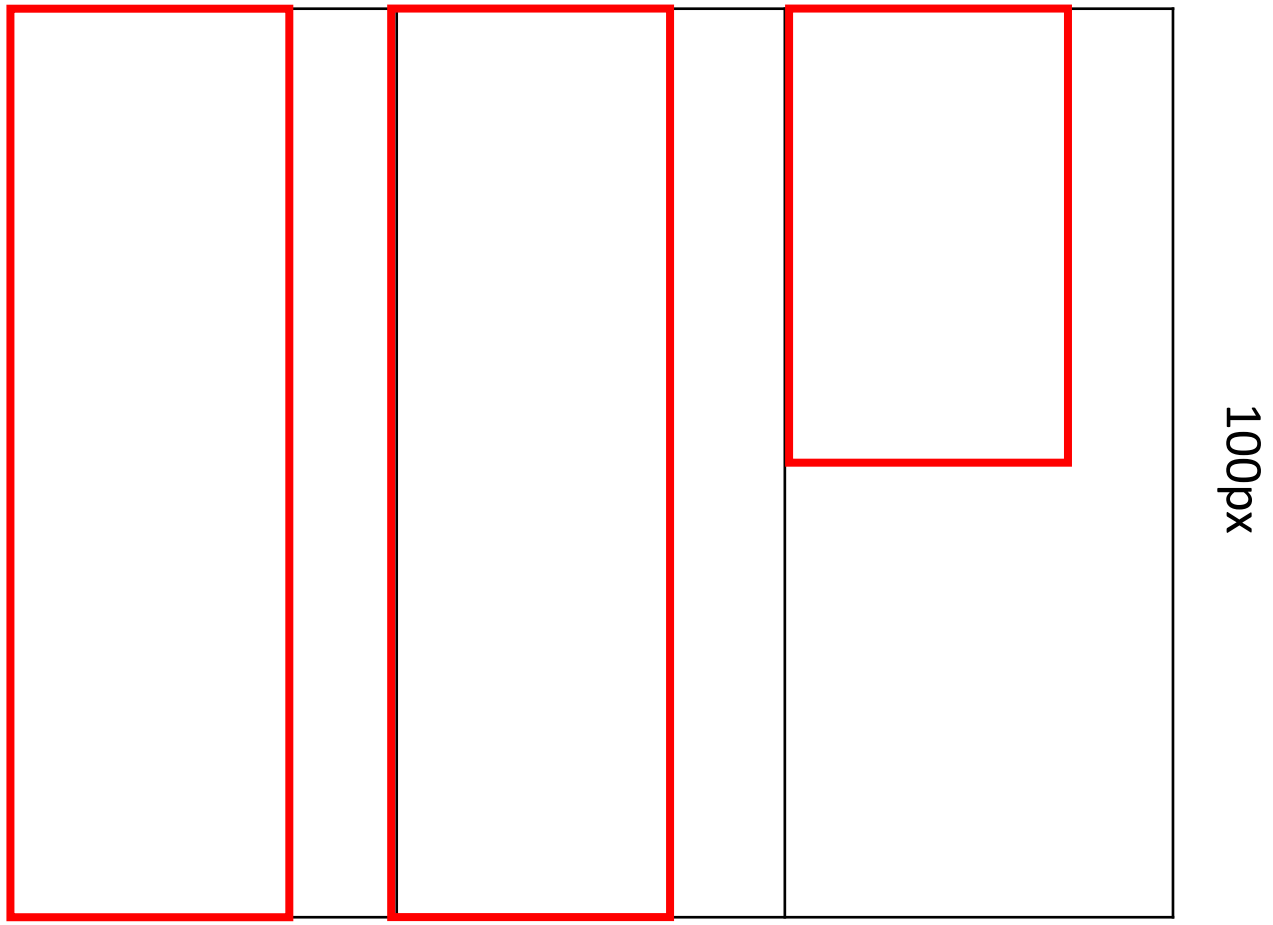
Example: height:auto; bottom:0;

```
.multicol {
  column-count: 3;
  height: 100px;
}
```



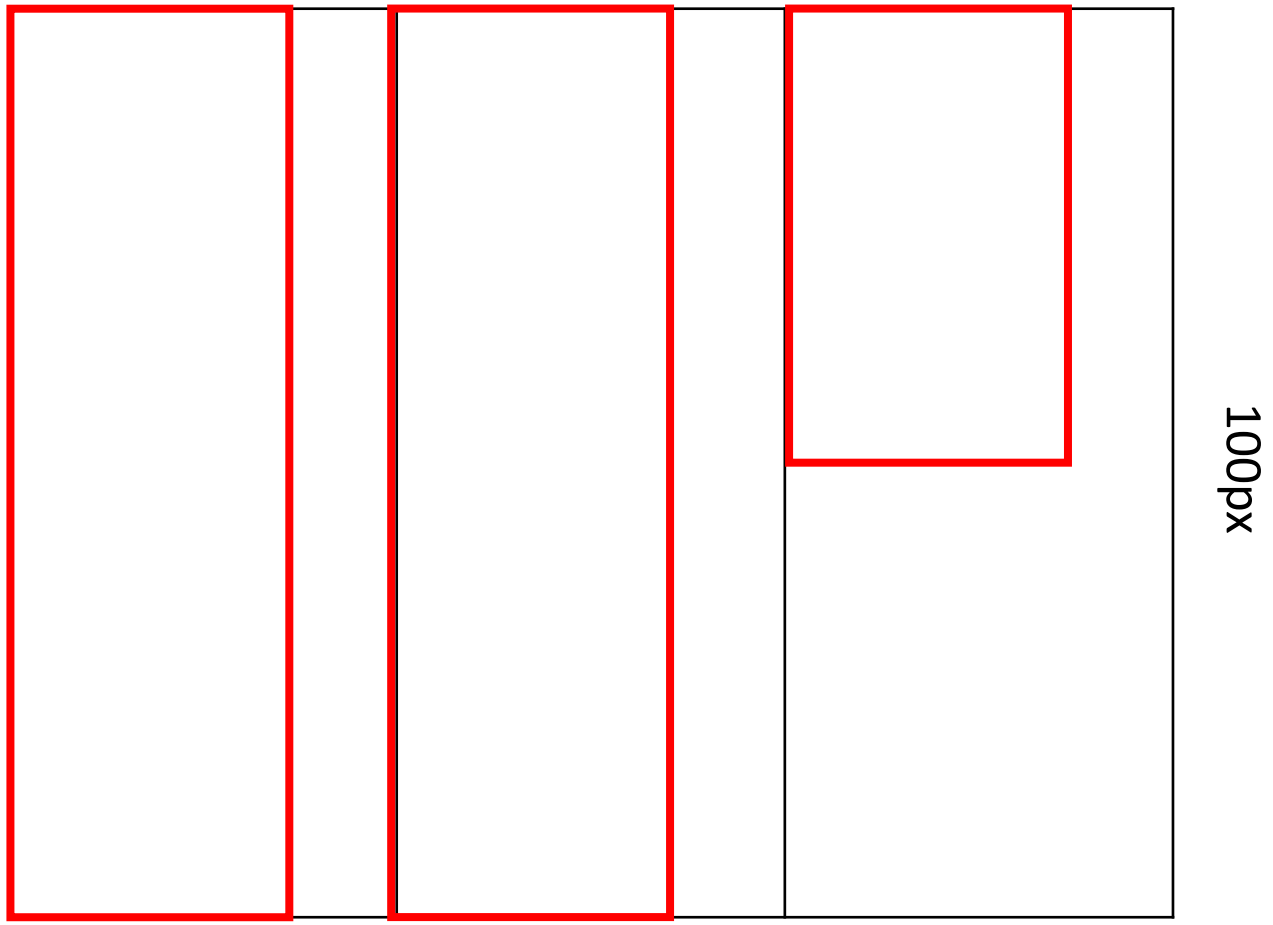
Example: height:auto; bottom:0;

```
.multicol {
  column-count: 3;
  height: 100px;
}
#rel {
  position: relative;
  height: 250px;
  outline: solid red;
}
```



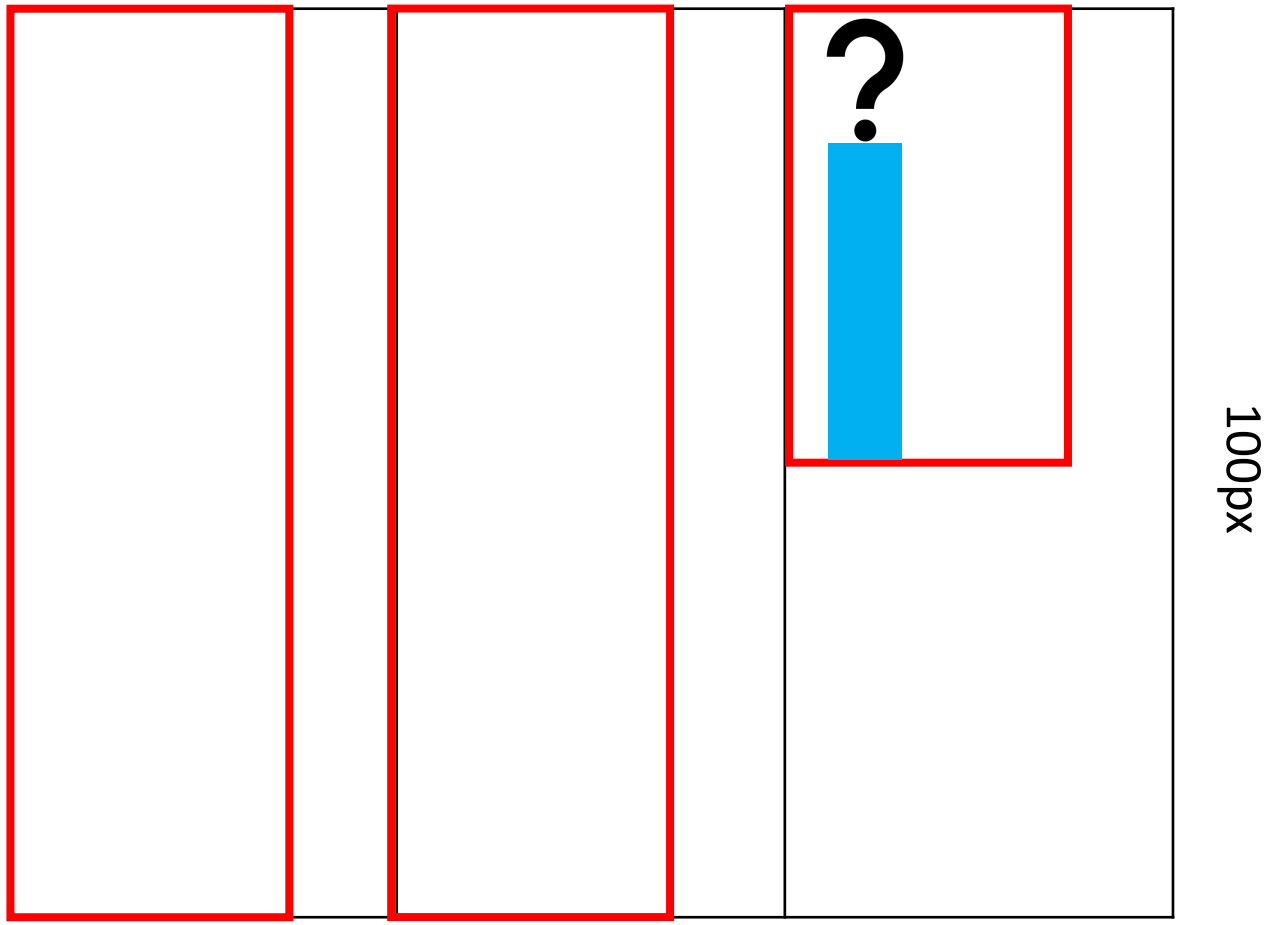
Example: height:auto; bottom:0;

```
.multicol {
  column-count: 3;
  height: 100px;
}
#rel {
  position: relative;
  height: 250px;
  outline: solid red;
}
#abs {
  position: absolute;
  height: auto;
  bottom: 0;
}
```



Example: height:auto; bottom:0;

```
.multicol {
  column-count: 3;
  height: 100px;
}
#rel {
  position: relative;
  height: 250px;
  outline: solid red;
}
#abs {
  position: absolute;
  height: auto;
  bottom: 0;
}
```



Example: height:auto; bottom:0;

- Run an additional non-fragmenting layout pass
- This will give us the block size -> start offset
- Then we can perform layout

Nested Fragmentation

Nested Fragmentation

** ** OutOfFlowPositionedNode 

```
    <div style="position: relative;">
        <div>
            <div style="position: absolute; bottom:0;"></div>
        </div>
    </div>
</div>
```

Nested Fragmentation

```
<div style="columns: 2;">
  <div style="columns: 3;">
    <div style="position: relative;">
      <div>
        <div style="position: absolute; bottom:0;"></div>
      </div>
    </div>
  </div>
</div>
```

OutOfFlowPositionedNode

Nested Fragmentation

```
<div style="columns: 2;">
  <div style="columns: 3;">
    <div style="position: relative;">
      <div>
        <div style="position: absolute; bottom:0;"></div>
      </div>
    </div>
  </div>
</div>
```

OutOfFlowPositionedNode

Nested Fragmentation


```
<div style="columns: 2;">
  <div style="columns: 3;">
    <div style="position: relative;">
      <div>
        <div style="position: absolute; bottom:0;"></div>
      </div>
    </div>
  </div>
</div>
```

Nested Fragmentation

OutOfFlowPositionedNode

```
<div style="columns: 2;">
  <div style="columns: 3;">
    <div style="position: relative;">
      <div>
        <div style="position: absolute; bottom:0;"></div>
      </div>
    </div>
  </div>
</div>
```

Nested Fragmentation

Design TBD

Contributing Team

Rossen Atanassov
Benjamin Beaudry
Morten Stenshorne
Ian Kilpatrick
Alison Maher
Daniel Libby

Questions?