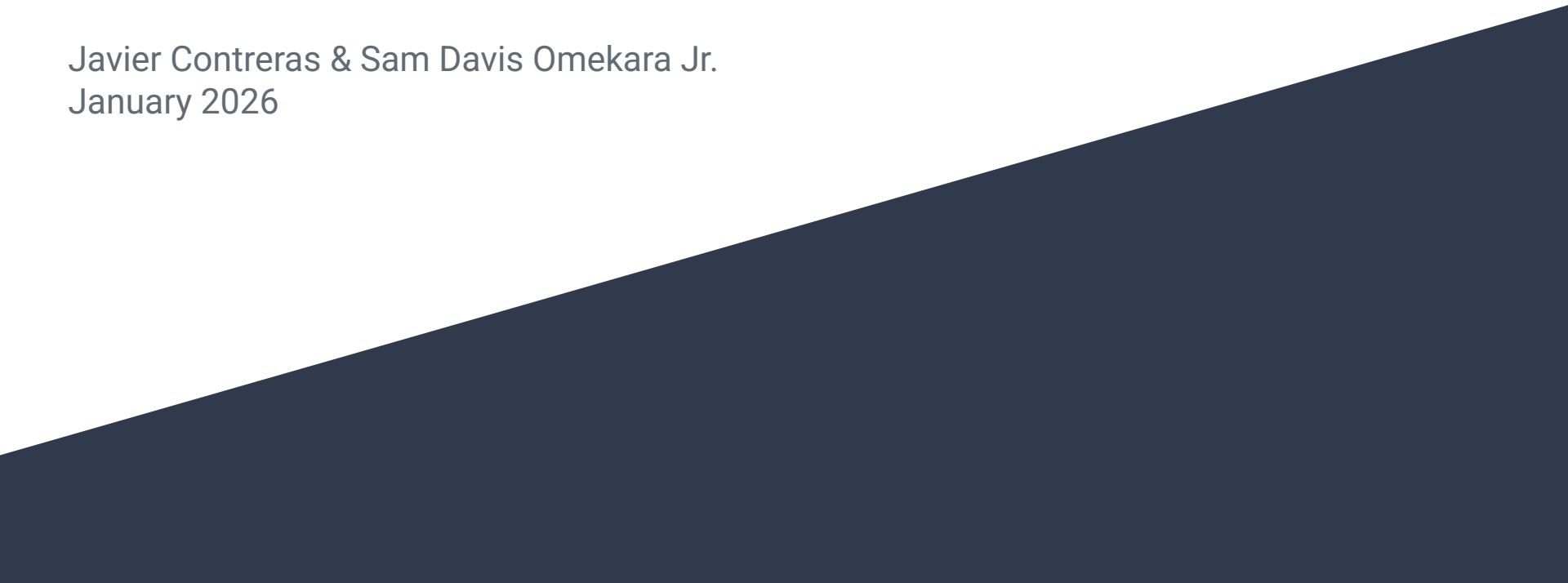


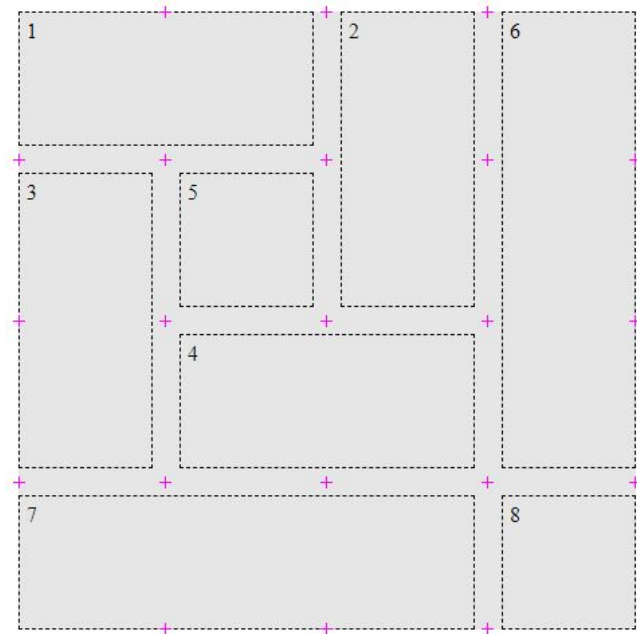
CSS-Gaps F2F

Javier Contreras & Sam Davis Omekara Jr.
January 2026

A dark blue diagonal gradient bar that starts from the bottom left corner and extends towards the top right corner, covering the lower half of the slide.

Intersections -> Segments

#13366

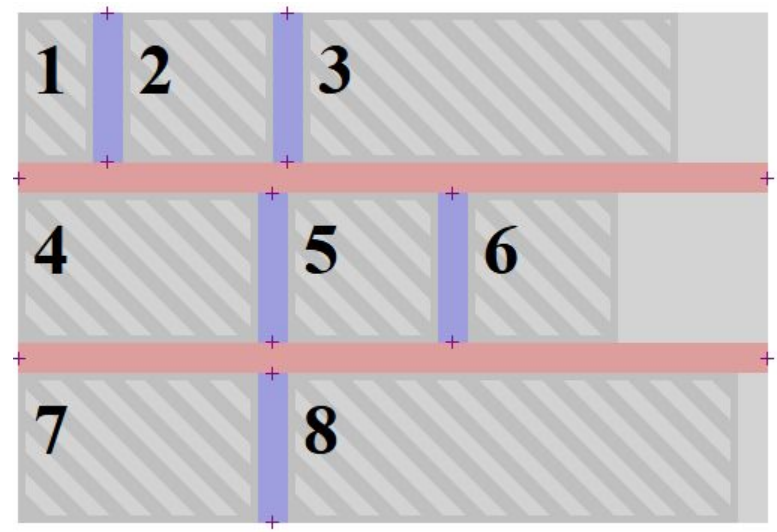
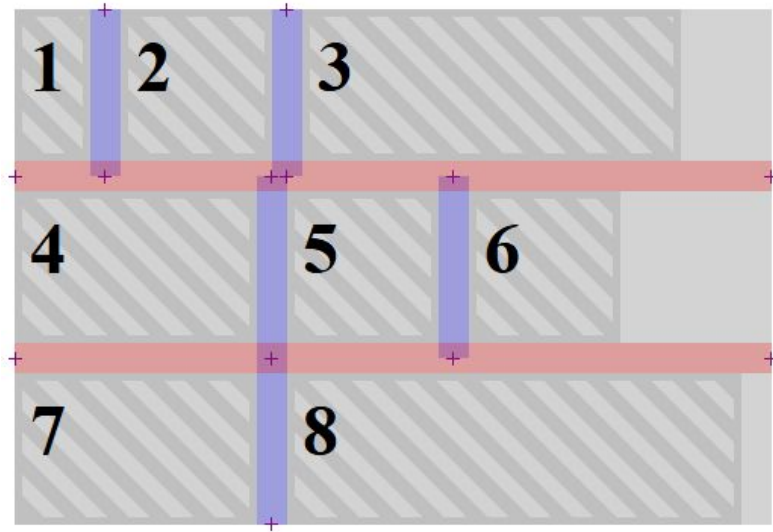


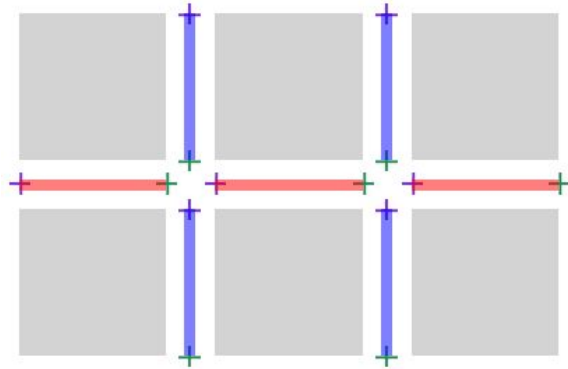
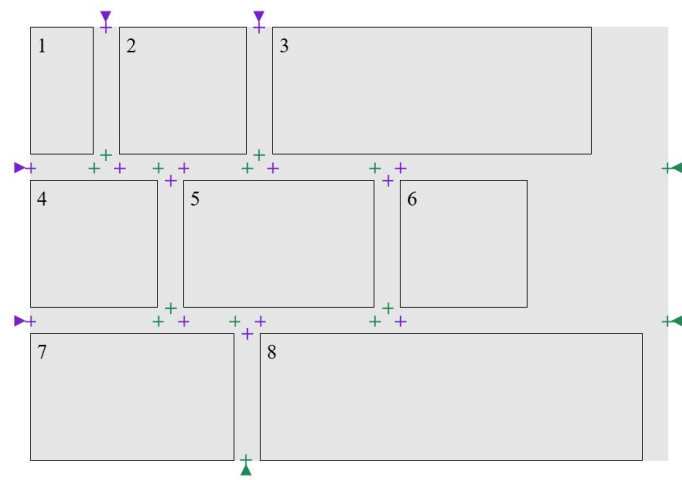
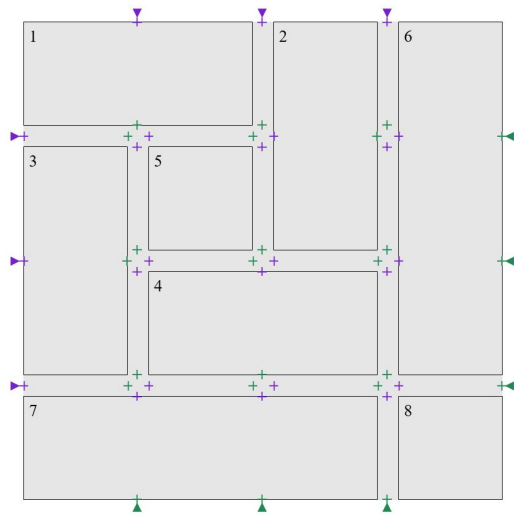
Currently, intersections are defined as:

- Center of the start or end edge of a gap
- Center of an **intersection** between gaps in different directions

Very Grid-like, does not scale to other containers

Intersections -> Segments (cont'd) #13366





- Segment endpoints are created:
- Bounds of a Gap
 - Where two gaps touch

Row decoration breaks in multicol

#13135

Do we want to allow decorations to break in the row direction for multicol?

Default:

Screens flickered to life across the workspace, each one displaying a jumble of logs, graphs, or half-written code. Someone had left a sticky

note on the keyboard reminding the team that the deployment window was closing soon, though no one seemed especially hurried. The network

graph spiked for a moment, dipped again, and settled into a steady heartbeat. Somewhere in the background, a cooling fan kicked up to a louder

alive. The morning light drifted slowly across the room, climbing the walls like it had all the time in the world. Outside, a few early risers were already

making their way down the street, each wrapped in their own thoughts and quiet routines. Inside, the space remained still, filled with the gentle hum of

appliances and the occasional creak of settling floorboards. None of it was particularly important, but together these small, ordinary

Allowing breaks:

Screens flickered to life across the workspace, each one displaying a jumble of logs, graphs, or half-written code. Someone had left a sticky note on the keyboard

reminding the team that the deployment window was closing soon, though no one seemed especially hurried. The network graph spiked for a moment,

dipped again, and settled into a steady heartbeat. Somewhere in the background, a cooling fan kicked up to a louder gear, as if announcing it was also

slowly across the room, climbing the walls like it had all the time in the world. Outside, a few early risers were already making

their way down the street, each wrapped in their own thoughts and quiet routines. Inside, the space remained still, filled with the gentle

hum of appliances and the occasional creak of settling floorboards. None of it was particularly important, but together these small,

Potential readability / flow concerns.

However, by not allowing the ability for row decorations to break, we create a **disparity** with all the containers, which do allow it.

Proposal: Allow authors to break decorations in the row direction (rule-break: intersection), but don't break by default (rule-break: none).

rule-break: auto

#13127

Currently the default value for `rule-break` is `spanning-item`. We are considering the following instead:

Grid:

- `rule-break: auto`
 - `column-rule-break: auto` → `spanning-item`
 - `row-rule-break: auto` → `spanning-item`

Flex:

- `rule-break: auto`
 - `column-rule-break: auto` → `none`
 - `row-rule-break: auto` → `none`

Multicol:

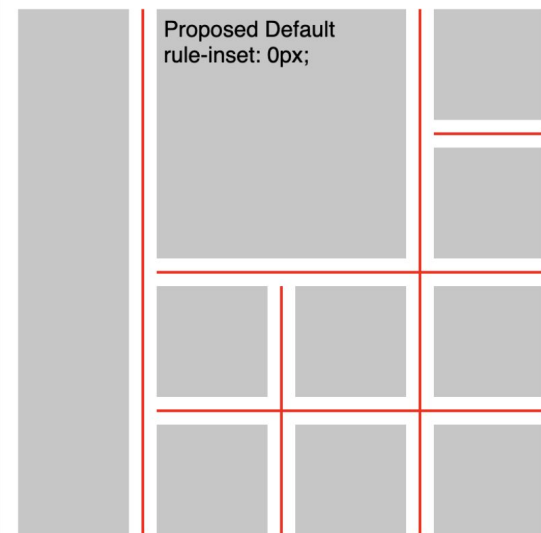
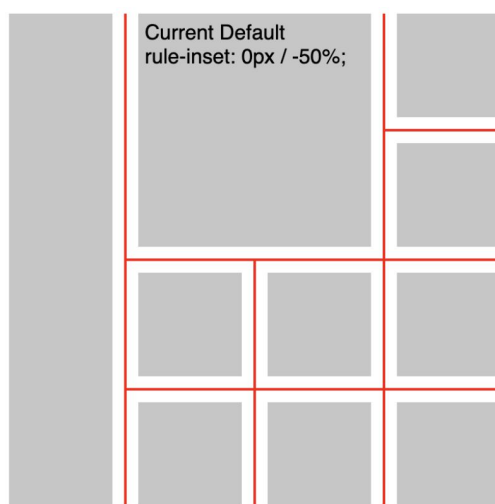
- `rule-break: auto`
 - `column-rule-break: auto` → `none`
 - `row-rule-break: auto` → `none`

Introducing **auto** and setting it as the default gives us flexibility for the future, and aligns with the fact that different containers have different sensible defaults.

Proposal: Introduce rule-break: auto as default, and resolve it to the values in the previous slide.

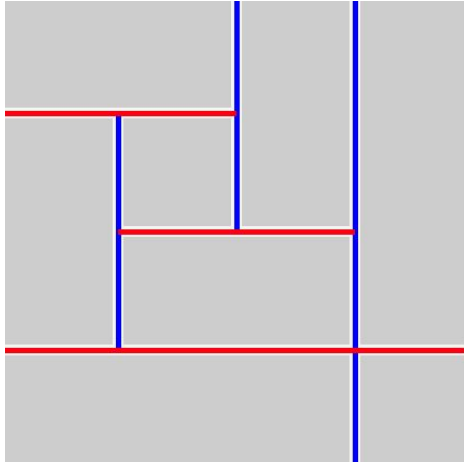
Insets initial value: 0

#13137

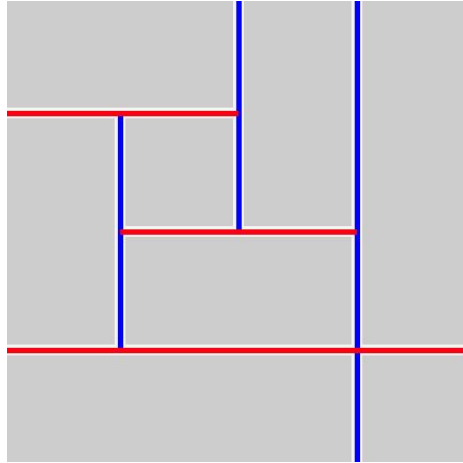


Currently, unless the rule-inset property is set to something other than the default (-50%), changing between the rule-break values does **not** change anything visually.

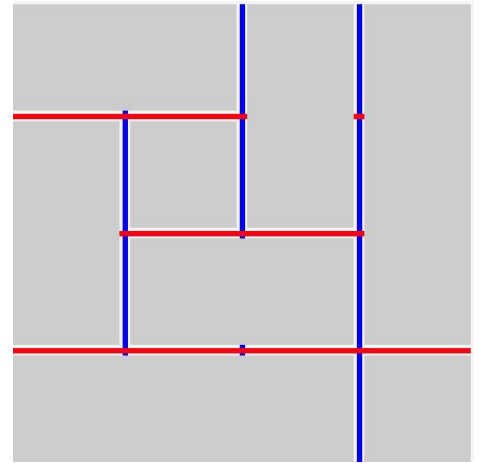
rule-break: intersection



rule-break: spanning-item

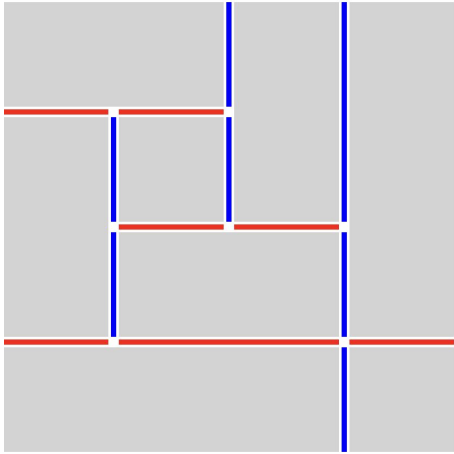


rule-break: none

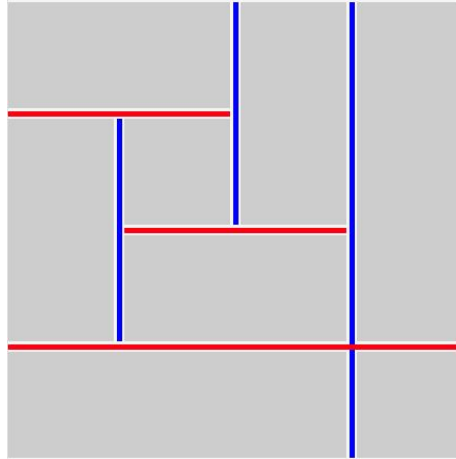


However, if the initial value is 0:

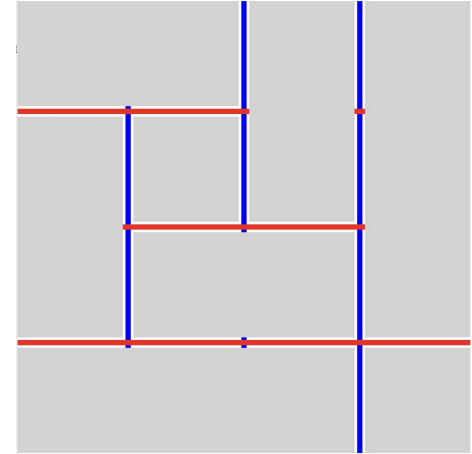
rule-break: intersection



rule-break: spanning-item



rule-break: none



Proposal: rule-inset initial value is 0 instead of -50%

#13137

Suppression of gaps across fragment breaks with spanning item

#13362

From [#11520](#)



css-meeting-bot on Jan 31, 2025

Member ...

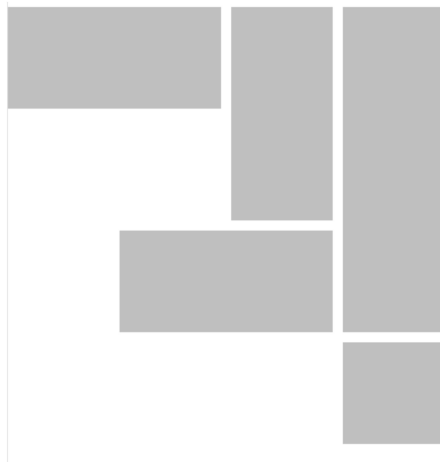
The CSS Working Group just discussed [css-grid-2] Suppression of gaps/gutters across fragment breaks , and agreed to the following:

- RESOLVED: if the gap overlaps a page break, or is the last content before a break, suppress it (across all container types)

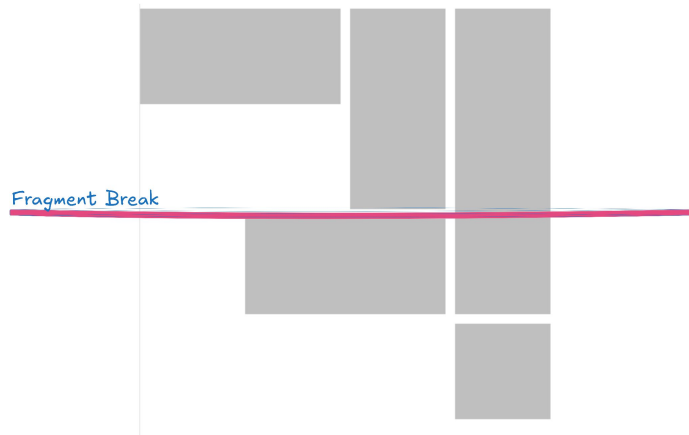


Suppression of gaps across fragment breaks with spanning item

#13362 (cont'd)

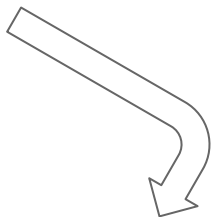


Caveat: A grid item might span a gap that is split across a break



Fragment Break

Problem: If we suppress gap, we might suppress a chunk of the item



Candidate gap for suppression

Spanning item occupying such gap

Proposal: Add an exception so that gaps are not suppressed when a spanning item covers the gap.

#13362

Space from content distribution as gutter

#12922

Typically: gutter \leq row-gap or column-gap

However,

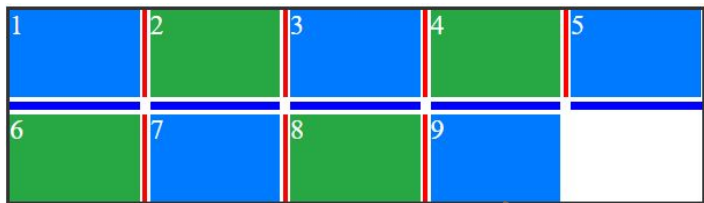
[CSS Align Level 3](#):

“Gutters effect a minimum spacing between items: additional spacing may be added by [justify-content/align-content](#). Such additional space **effectively increases the size of these [gutters](#).**”

- Tension:
 - Percentage insets which resolve against the gap in the cross direction.
 - Possibility of a case where the gutters in a given direction have differing widths over a given container

% insets most common use case is:
-50%

```
rule-inset: 0px;
```

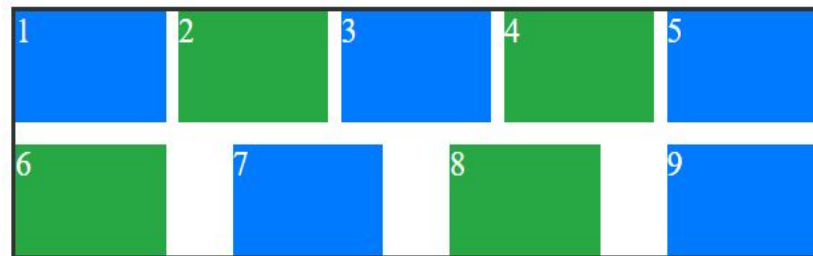


```
rule-inset: -50%;
```



Varying gutter sizes

```
display: flex;  
flex-wrap: wrap;  
justify-content: space-between;
```



Proposal: alignment space == gutters; %insets => `join`

#13137

rule-break naming

#11491

Currently we have:

- rule-**break**: none
- rule-**break**: spanning-item
- rule-**break**: intersection

With “segments” rather than “intersections”-based decorations, the word “break” makes less sense, since “segments” are already “**broken**”.

We could rename to something like:

- rule-break: none → rule-**join**: all
- rule-break: intersection → rule-**join**: none
- rule-break: spanning-item → rule-**join**: avoid-spanning-item???

However, rule-break could simply be **reinterpreted**.
Under this approach, we can define rule-break as applying to **decorations**, not to the internal segment structure. The distinction becomes:

Segments are the mechanism
Decorations are the painted artifact

Proposal: Keep rule break naming as is, just change spec text around it.

#13362