Div with `im_merge`

Merge subsequent `Image`'s inside a `Div` with class `im_merge`.

Due to a *feature request* by *pbsds* `pandoc-imagine` has been extended to better cooperate with *pandoc-crossref*, whose *subfigure grids* facility requires that consecutive image links be located inside a single paragraph.

This means that `Div`'s, when assigned the `pandoc-imagine` specific class `im_merge`, will have their block-level elements processed individually. Any consecutive `Image`-links are collected into a single `Para`, other elements are included in the `Div` contents as-is.

A processed `CodeBlock` (inside such a `Div`) may yield a list of block-level results via the `im_out="x,y,z,.."` option, so any and all `Image`'s in the result of a processed `CodeBlock` are extracted and appended to the previous `Para` if possible. The remainder, if any, will be appended to the `Div`'s list of block-level elements. If that happens, it will breakup the inlining of `Image`'s, so its best to not use the `im_out` option in such `im_merge-Div`'s.

Example `Div`

The following sections basically use the following `Div`, each time slightly modified to show differences in results and/or features of `im_merge`.

````
:::: { #inlined .im_merge }

```{.dot width=25% #img1 caption="ONE"}
digraph {
  node [style=filled fontcolor=black fillcolor=red margin=0 fontsize=20];
  A -> a;
}
```

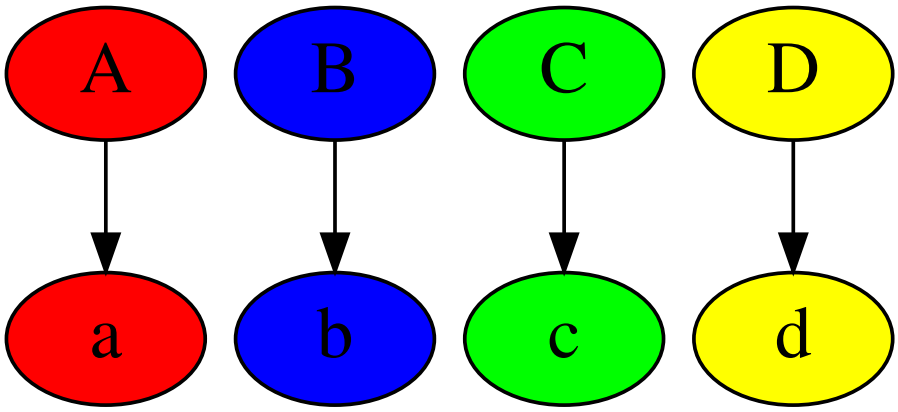
```{.dot width=25% #img2 caption="TWO"}
digraph {
  node [style=filled fillcolor=blue margin=0 fontsize=20]
  B -> b;
}
```

```{.dot width=25% #img3 caption="THREE"}
digraph {
  node [style=filled fillcolor=green margin=0 fontsize=20]
  C -> c;
}
```

```{.dot width=25% #img4 caption="FOUR"}
digraph {
  node [style=filled fillcolor=yellow margin=0 fontsize=20]
  D -> d;
}
```

End of Div
::::
````

yields



End of Div

Notes:

- `CodeBlock`'s in an `im_merge`-Div, may be separated by 1+ empty lines
- Captions are gone, since only `Image`'s in a paragraph of their own, get a caption. From the *manual*:

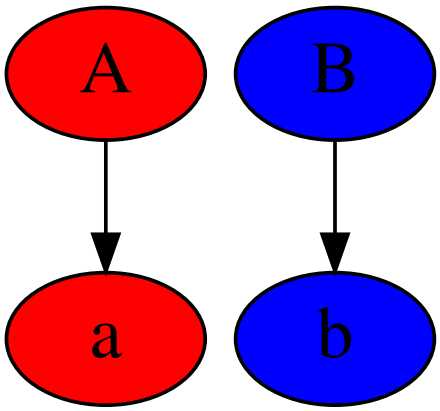
  An image with nonempty alt text, occurring by itself in a paragraph, will be rendered as a figure with a caption. The image's alt text will be used as the caption.

  How this is rendered depends on the output format. Some output formats (e.g. RTF) do not yet support figures. In those formats, you'll just get an image in a paragraph by itself, with no caption.

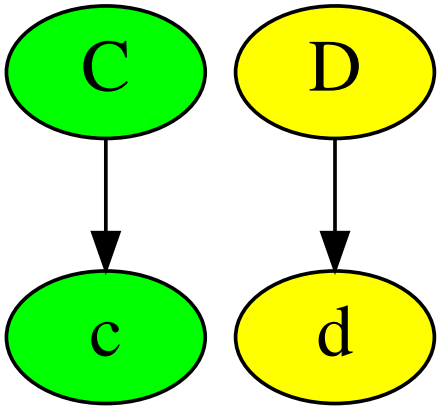
- Reportedly, *pandoc-crossref*, seems to handle that separately.

Only consecutive Images are `im_merged`

If text is inserted in between `CodeBlock`'s, it'll breakup the merging. Below is the output of the same `Div`, but with some text inserted between the 2nd and 3rd `CodeBlock`.



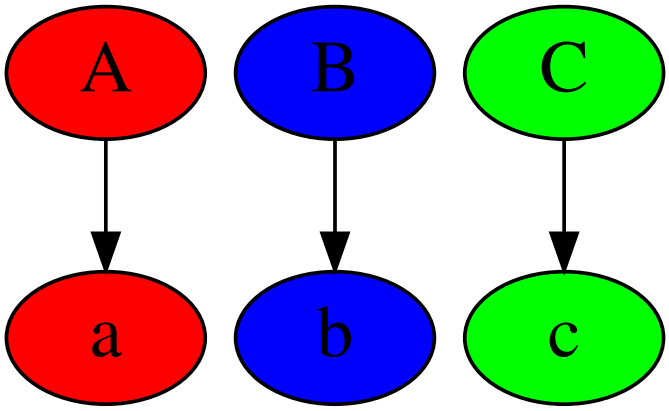
Text inserted between 2nd and 3rd `CodeBlock`.



End of Div

Images with `im_out` options

Again, using the original `Div` shown at the top, but now the 3rd Image has the option `im_out="img,fcb"`, which breaks the merging of Images into a single `Para`. Since the 4th `CodeBlock` represents the last `Image`, it is contained in a `Para` of its own, which makes the caption show up.




````
```{#img3 .dot width="25%" im_out="img,fcb" caption="THREE"}
digraph {
  node [style=filled fillcolor=green margin=0 fontsize=20]
  C -> c;
}
```
````

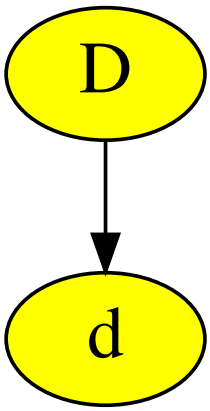
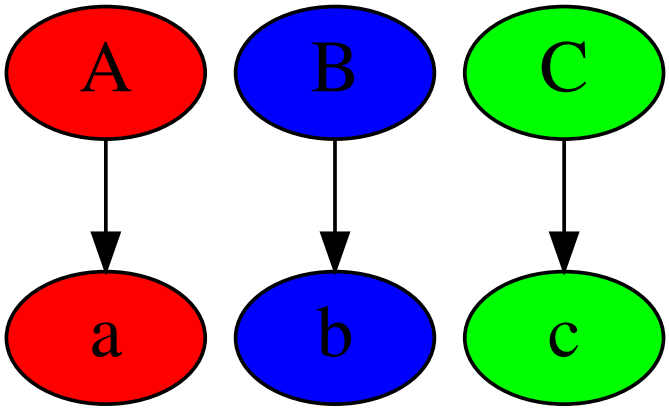


Figure 1: FOUR

End of Div

Swapping the order of the outputs of the 3rd `CodeBlock`, i.e. `im_out="fcb,img"`, makes no difference.




```
```{#img3 .dot width="25%" im_out="fcb,img" caption="THREE"}
digraph {
  node [style=filled fillcolor=green margin=0 fontsize=20]
  C -> c;
}
```
```

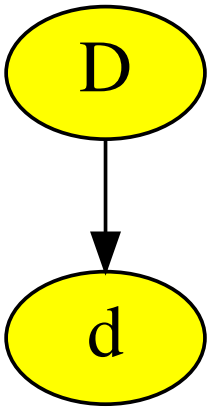
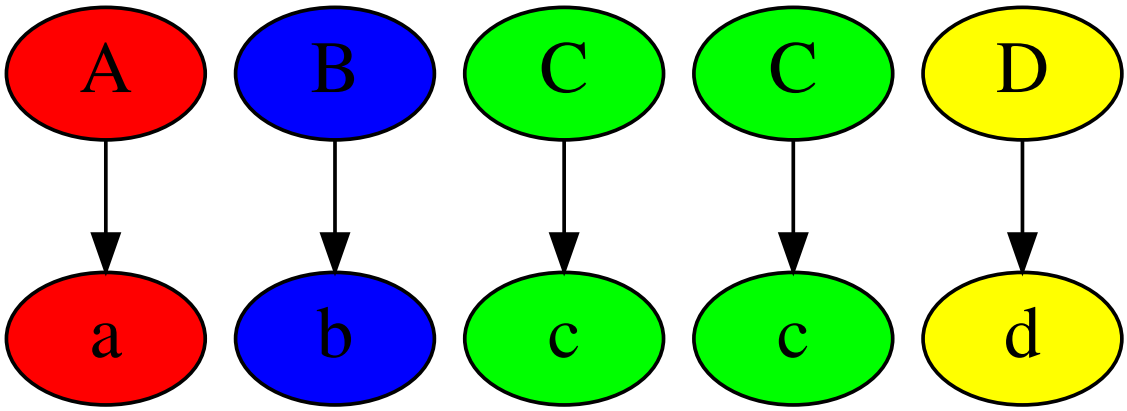


Figure 2: FOUR

End of Div

Repeat an Image with im_out

Imagine will produce its output(s) according to the `im_out` option that specifies what types of output to produce and in which order. During inlining of `Image`'s, any repeated images of 1 codeblock are all extracted and inlined. Setting the `im_out=img,img` of the 3rd `CodeBlock`, yields:



End of Div

In the processed `Div` the 3rd `CodeBlock` was modified to:

````
```{.dot width=25% #img3 im_out=img,img caption="THREE"}
digraph {
  node [style=filled fillcolor=green margin=0 fontsize=20]
  C -> c;
}
```
````

A Div without class `im_merge` is handled normally

Again using the original `Div`, but now without the `im_merge` class attached to it, so it is processed per normal. Each `CodeBlock` now yields an `Image` of its own in its own `Para`, which makes the captions show up this time.

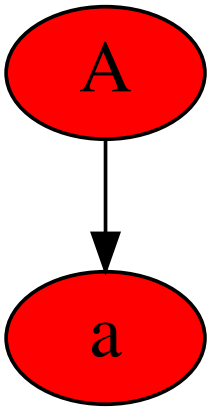


Figure 3: ONE

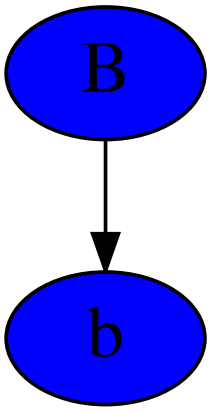


Figure 4: TWO

End of Div

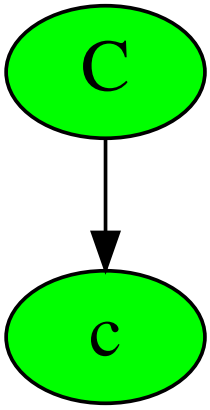


Figure 5: THREE

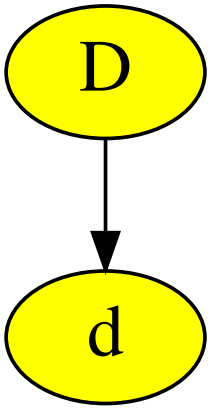


Figure 6: FOUR