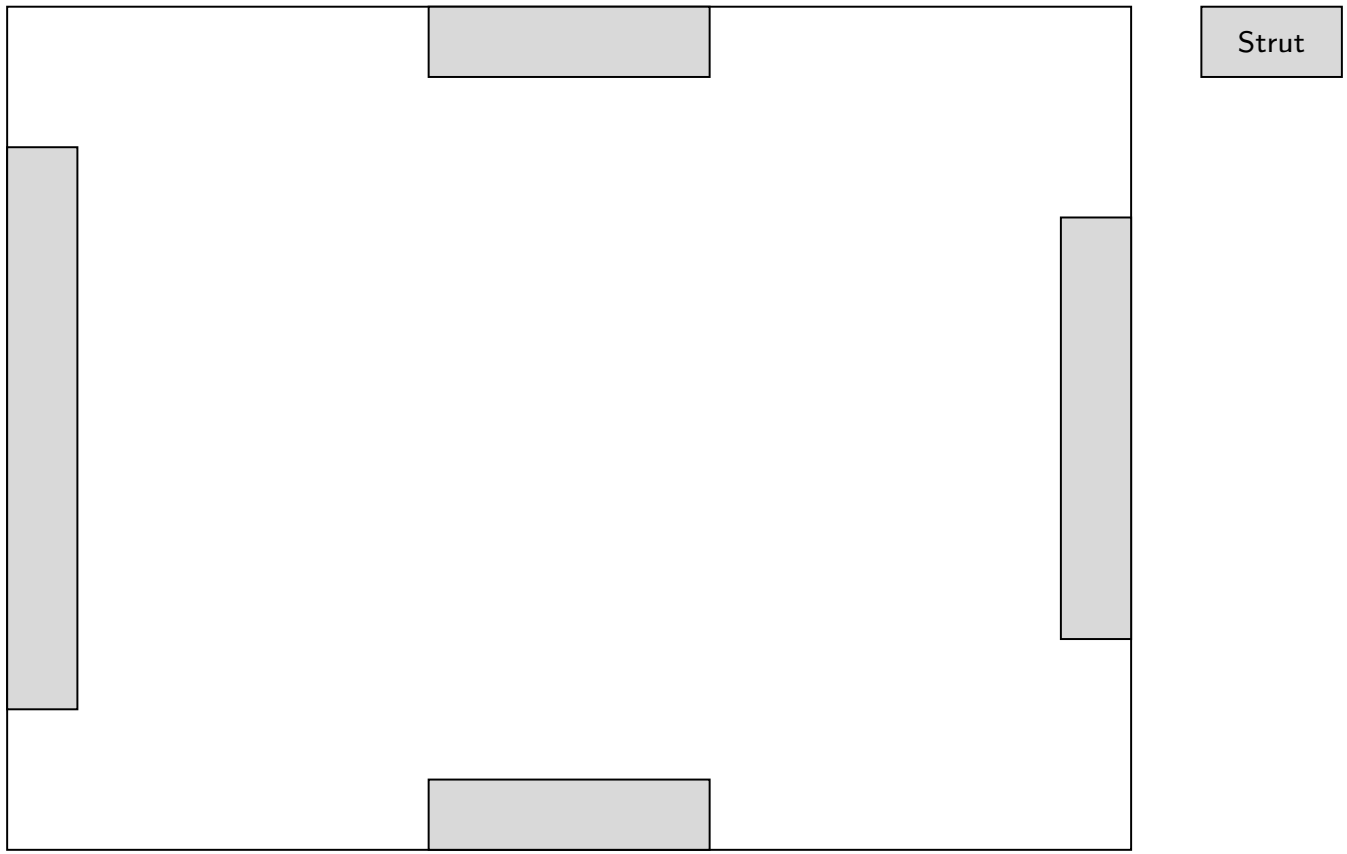
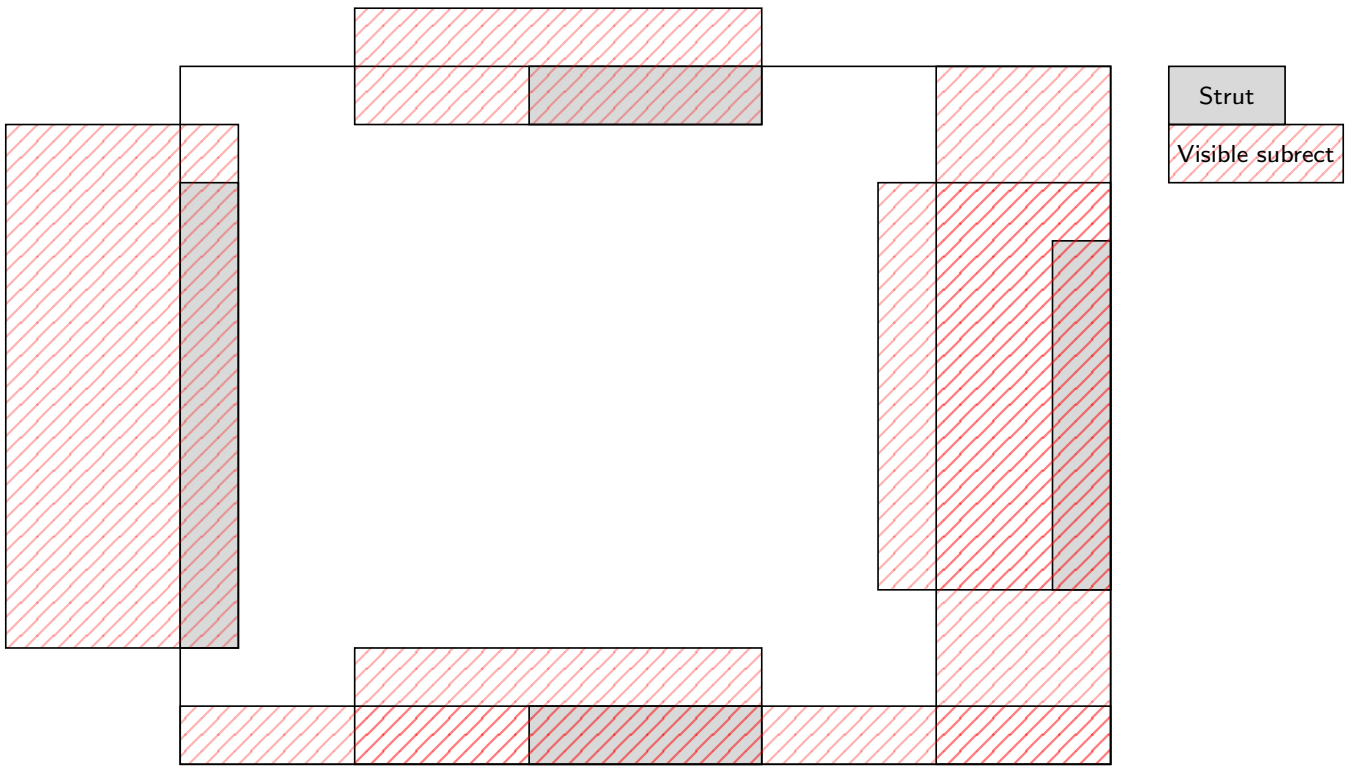
moveResize restriction algorithm

February 1, 2025

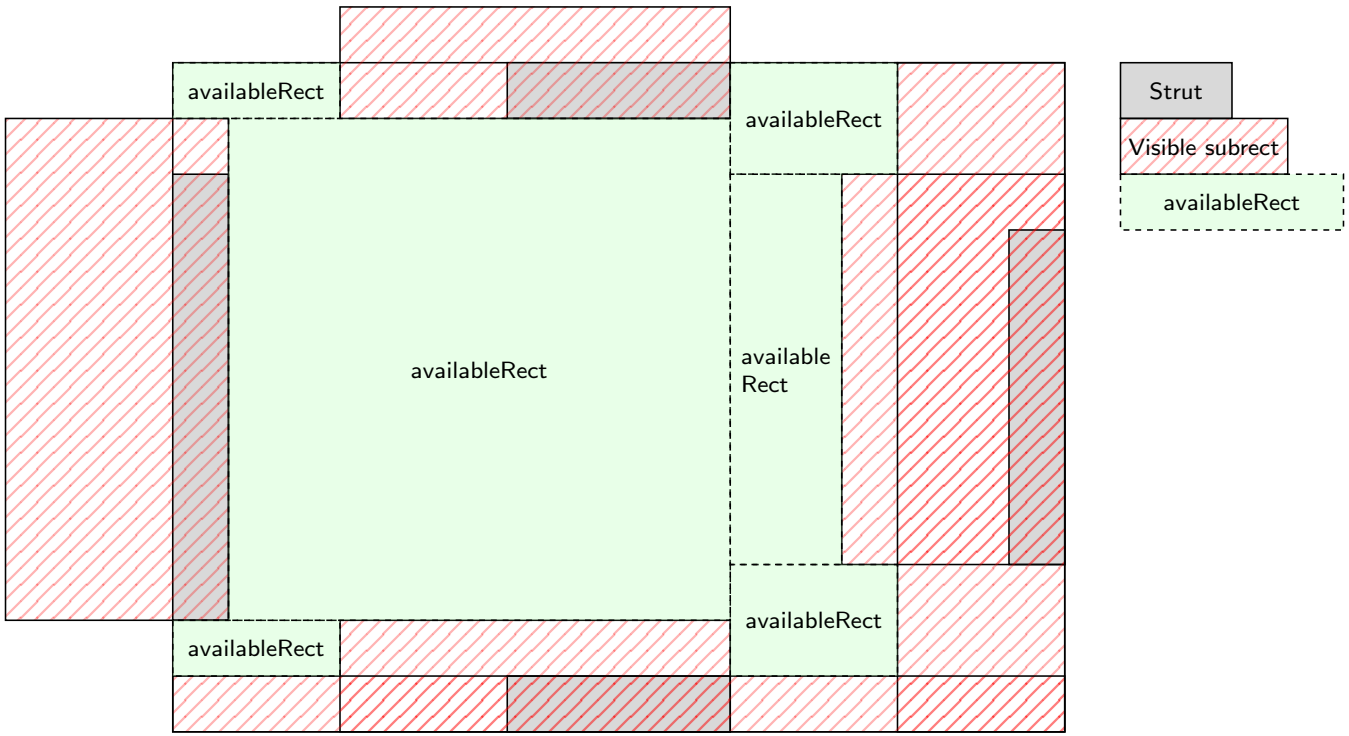
The big rectangle represents the overall available area – windows are not visible outside. The grey rectangles represent the struts (think of them as obstacles – windows are not visible in these areas).



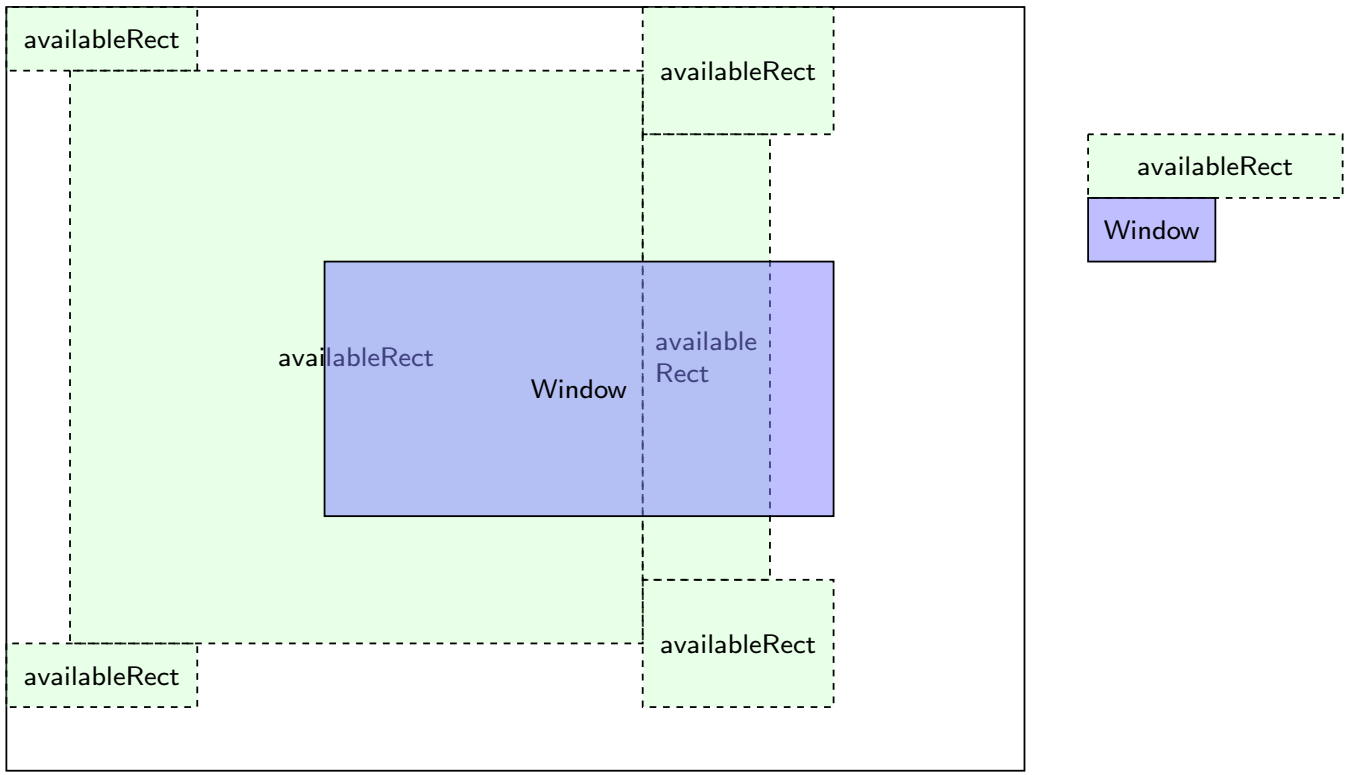
Since the titlebar need to have certain number of continuous visible pixels, extend each strut to the left (by requiredPixels) and to the top (by titlebarHeight). Shrink the overall available area from the right and bottom by the same amount. These are the areas where the top left corner of the visible titlebar subrect cannot be placed (red diagonal lines). The remaining white area is availableRegion.



Since availableRegion is a QRegion, it is automatically split into rectangles (green with dashed borders, and “availableRect” in center).



The next step depends on window location (shown in blue) and move/resize type.

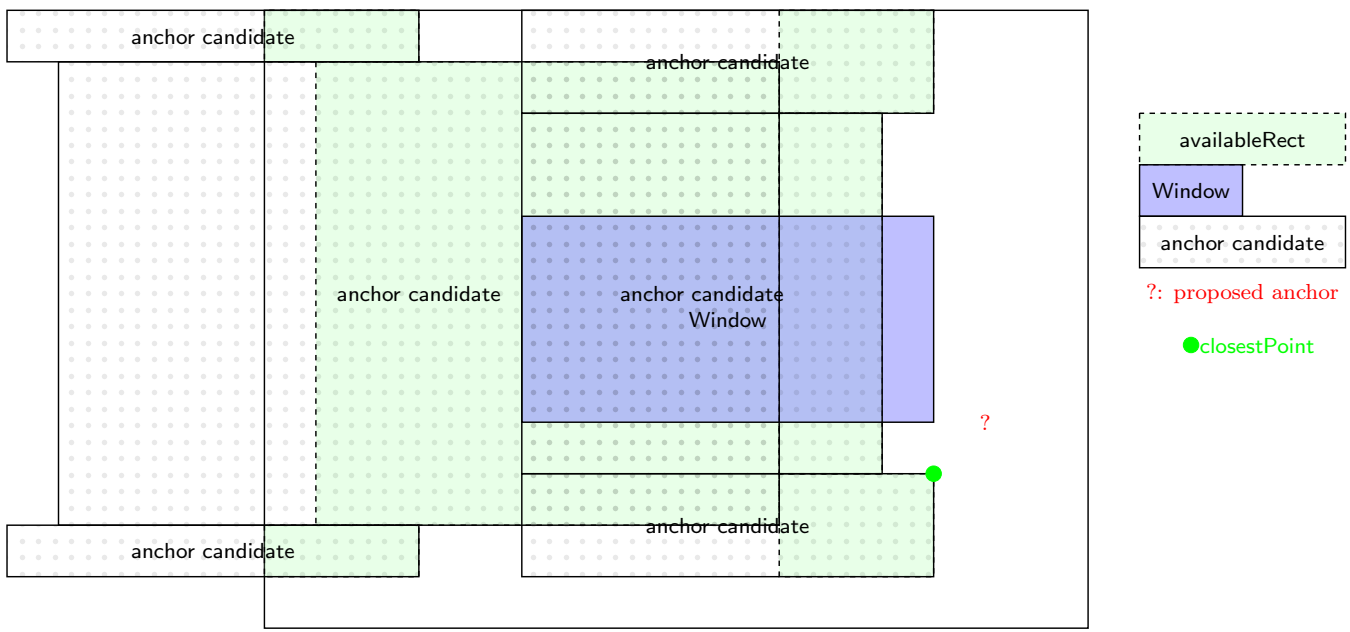


Move

Assume *move* for now. Recall availableRect stores possible locations of the top left corner of the visible titlebar subrect. For each availableRect (stopping early if availableRect becomes empty):

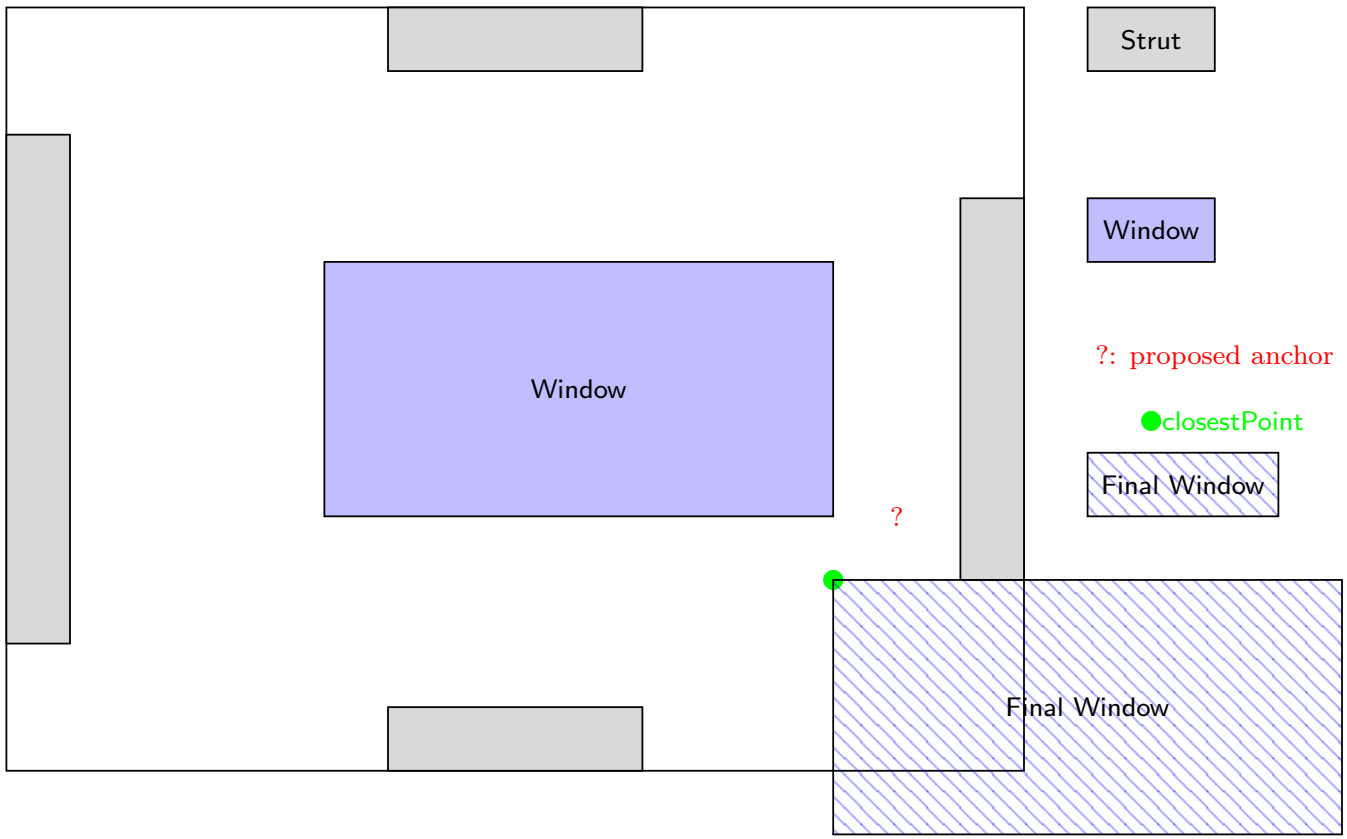
- Apply restrictions to visible subrect top-left
  - None needed for move.
- Convert visible subrect top-left to window top-left
  - Extend each availableRect to the left by windowWidth - requiredPixels (gray dots, with text anchor candidate).

For proposed anchor point (top left corner of the window, calculated from user input, red question mark), find closest anchor candidate point (green circle).



Move

We can visually inspect the solution.

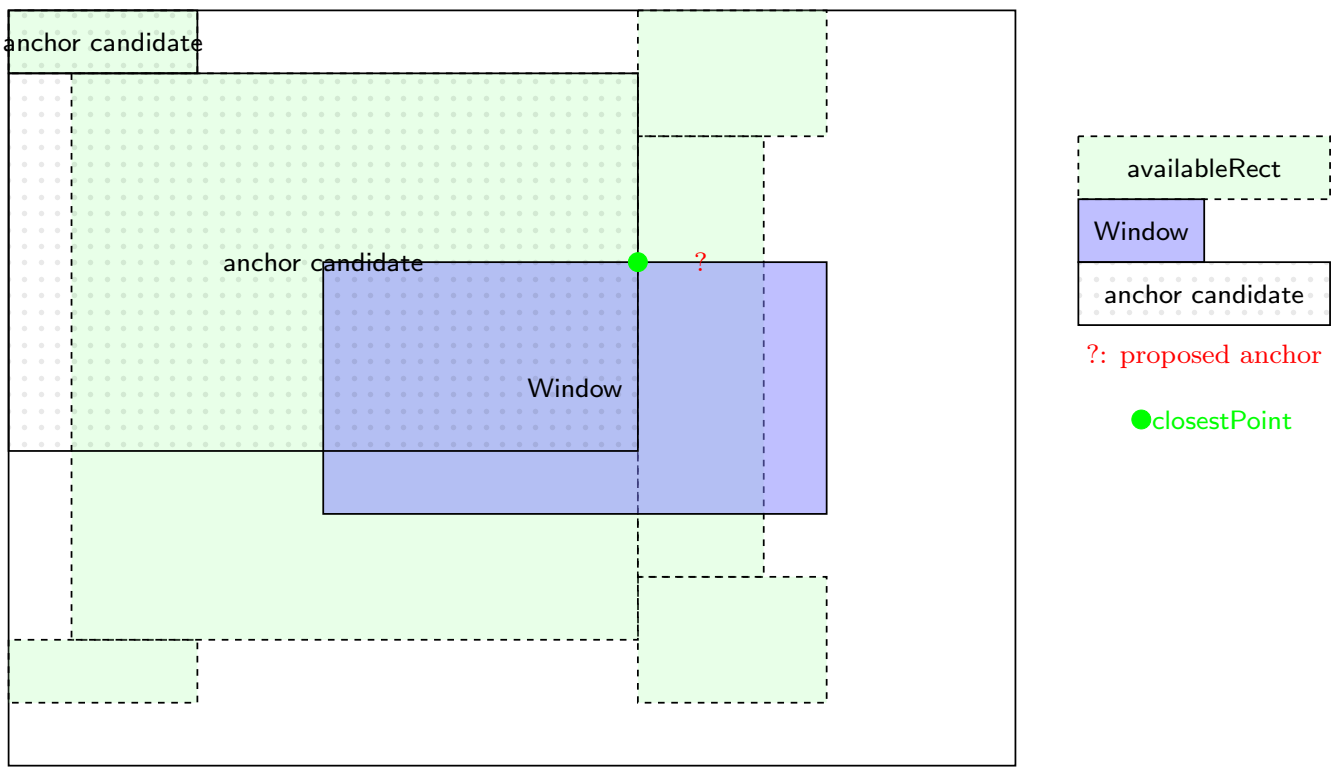


Resize Left

Now assume user is resizing the *left* of the window. For this case anchor is also top left. For each availableRect (stopping early if availableRect becomes empty):

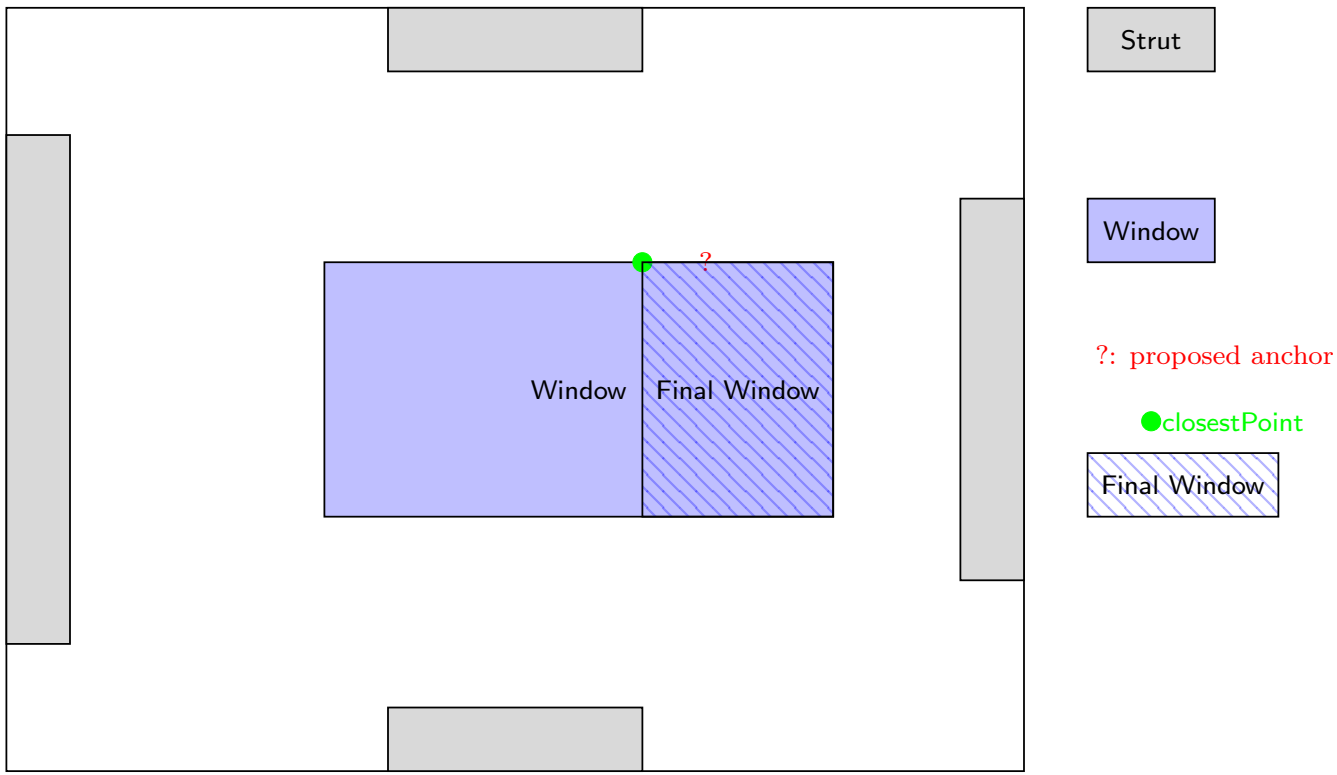
- Apply restrictions to visible subrect top-left
  - clip bottom to windowBottom - titlebarHeight (always performed for resize);
  - clip right to windowRight - requiredPixels;
- Convert visible subrect top-left to window top-left
  - extend left to overall available area left.

For proposed anchor point (red question mark), find closest anchor candidate point (green circle), while only allowing horizontal movement.



Resize Left

We can visually inspect the solution.

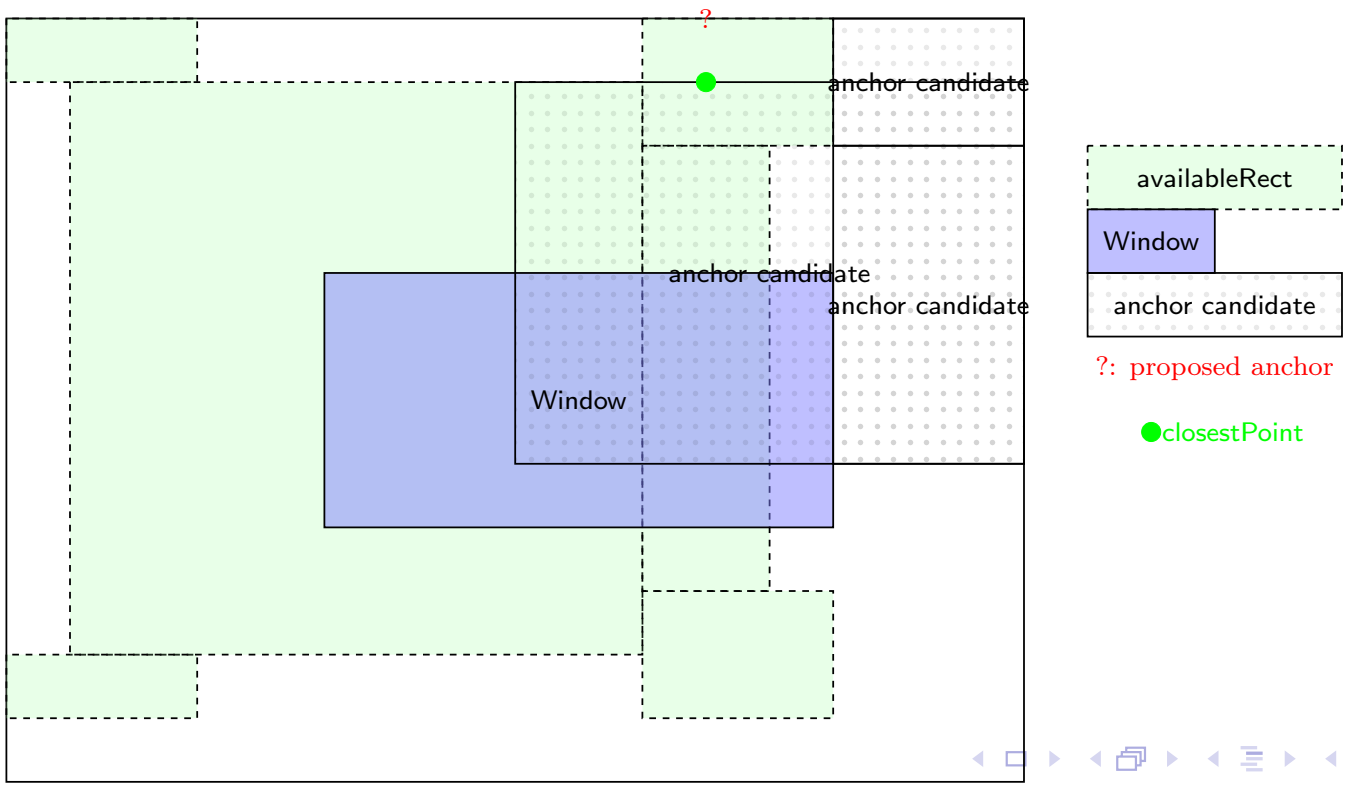


Resize Top Right

Now assume user is resizing the *top right* of the window. For this case anchor is top right. Transform availableRect and convert to possible locations of the top right corner of the window. For each availableRect (stopping early if availableRect becomes empty):

- Apply restrictions to visible subrect top-left
  - clip bottom to windowBottom - titlebarHeight (always performed for resize);
  - clip left to windowLeft (top-left of visible subrect must be right of windowLeft);
- Convert visible subrect top-left to window top-right:
  - extend right to overall available area right;
  - move left right by requiredPixels;

For proposed anchor point (red question mark), find closest anchor candidate point (green circle).



Resize Top Right

We can visually inspect the solution.