

**NOTE:** The following instructions were published on Confluence. This document was created in MS Word, reviewed, and then ported to a Confluence page.

# How To Create an Image Map for a Flowchart

Image mapping allows you to create clickable areas on an image, enabling users to interact with specific regions that link to different destinations, such as web pages or documents. For flowcharts, image mapping makes each step or node clickable, directing users to detailed instructions or related content.

The following instructions explain how to create an image map for a flowchart using HTML `<map>` and `<area>` tags, with Microsoft Paint as the tool to determine coordinates. The process involves identifying coordinates for clickable areas, defining their shapes, and linking them to appropriate URLs or resources, using the provided example code as a reference.

## Prerequisites

- A flowchart image (PNG file) with clearly defined regions and a width of no more than 800 pixels.
- Microsoft Paint (available on Windows) to determine coordinates.

## Create Image Maps

The following instructions work with flowcharts containing rectangles, squares, octagons, and circles.

1. Open the image in Microsoft Paint.
2. Determine the coordinates of each element to be mapped.
  - To determine the coordinates of a square, rectangle, or octagon, refer to Calculate “rect” Coordinates.
  - To determine the coordinates of a circle or oval, refer to Calculate “circle” Coordinates.
3. Use the following code example as a template for your image mapping. Copy and paste additional `<area>` lines, as needed. Modify the code to match the elements you calculated in the previous step.

```
<map name="diagramMap">
  <area shape="rect" coords="17,18,220,229" href="{% link [path]/example1.md %}" alt="alt text" title="title">
  <area shape="circle" coords="140,135,111" href="{% link [path]/example2.md %}" alt="alt text" title="title">
</map>
```

- a. Enter the **shape** of the clickable area. Values are “rect” or “circle”.
  - b. Replace the **coords** values to match the values you calculated in step 2.
  - c. Replace the **path** values with the path pointing to your Markdown file.
  - d. Enter the **alt** text associated with the mapped element.
  - e. Enter the **title** associated with the mapped element.
4. Copy the entire `<map>` code block and place it in your Markdown file immediately after your image (`<img>`) definition. The following image is an example of a final image mapping.

```


  <area shapes="rect" coords="3,13,93,61" href="{% link docs/onboarding/getting-started/laptop.md %}" alt="Step 1: Laptop" title="Go to Step 1: Laptop">
  <area shapes="rect" coords="127,2,218,72" href="{% link docs/onboarding/getting-started/vscode-config.md %}" alt="Step 2: VSCode Config" title="Go to Step 2: VSCode Config">
  <area shape="rect" coords="253,2,363,72" href="{% link docs/onboarding/getting-started/git-introduction.md %}" alt="Step 3: GIT Introduction" title="Go to Step 3: GIT Introduction">
  <area shape="rect" coords="398,2,508,72" href="{% link docs/onboarding/getting-started/torc-docker.md %}" alt="Step 4: Torc Docker" title="Go to Step 4: Torc Docker">
  <area shape="rect" coords="543,2,653,72" href="{% link docs/onboarding/getting-started/cloud-vdi.md %}" alt="Step 5: Cloud VDI" title="Go to Step 5: Cloud VDI">
  <area shapes="rect" coords="688,2,797,72" href="{% link docs/onboarding/getting-started/model-project.md %}" alt="Step 6: Model Project" title="Go to Step 6: Model Project">
</map>
```

## Calculate “rect” Coordinates

The following procedure describes how to calculate the coordinates for a square, rectangle, or octagon. While the shape is defined in the code as “rect” (rectangle), it can be applied to any element where a mapped right-angle assignment can be superimposed. These steps use Microsoft Paint to determine the coordinates.

Coordinates for these shapes require two sets of X,Y coordinates. The X determines the pixels across while the Y determines the pixels down. For example, if the top of a square begins 20 pixels to the left of the image and 20 pixels from the top, the X,Y coordinates are “20,20”.

1. Open the image in Microsoft Paint.
2. Hover the mouse arrow over the top-left of the region to be mapped. The X,Y coordinates are displayed on the bottom left of the GUI. The following images show the X,Y coordinates of the first flowchart block titled “rect”.



3. Write down the X,Y values. This set of values represents the top-left of the element. In this example, the X value is 19 and the Y value is 21. This is expressed as “19,22” in the code and will be used in a later step.
4. Hover the mouse arrow over the bottom-right of the region to be mapped.



5. Write down the X,Y values. This set of values represents the bottom-right of the element. In this example, the X value is 249, and the Y value is 252. This is expressed as “249,252” in the code.
6. Repeat steps 2 through 5 for each element that requires coordinates, documenting each set of coordinates.
7. For each mapped element in the image, use the following code to customize the coordinates you just calculated. Enter the coordinate values in the *coords* attribute, as shown in the following code. The values in this example represent the coordinates calculated in the previous steps. Place the top-left X,Y values with the bottom-right X,Y values. In the example above, this is expressed as “19,22,249,252”.

```
<map name="diagramMap">
  <area shape="rect" coords="19,22,249,252" href="{% link [path]/example1.md %}" alt="alt text"
  title="title">
</map>
```

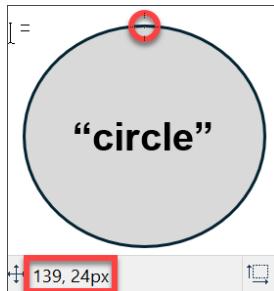
8. Copy the `<area>` code block and enter the coordinates for each element in your drawing.

## Calculate “circle” Coordinates

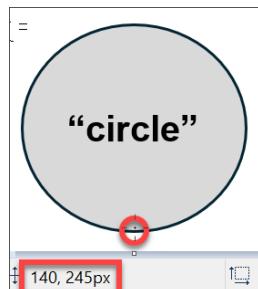
The following procedure describes how to calculate the coordinates for a circle or oval. These steps use Microsoft Paint to determine the coordinates.

Circle coordinates require calculating the radius and determining the center coordinates.

1. Calculate the radius of the circle (represented as  $r$  in the following calculations). The radius is the distance from the center of the circle to its edge, measured in pixels.
  - a. Move the cursor to the top edge of the circle and note the  $y_{top}$  coordinate. In this example, the coordinates are 139,24. The value 139 represents the x coordinate. The value 24 represents the y coordinate. Write down 24 as the  $y_{top}$  coordinate.



- b. Move the cursor to the bottom edge of the circle and note the  $y_{bottom}$  coordinate. In this example, the coordinates are 140,245. The value 140 represents the x coordinate. The value 245 represents the y coordinate. Write down 245 as the  $y_{bottom}$  coordinate.



- c. Calculate the diameter:  $diameter = y_{bottom} - y_{top}$ . Using the values in the example above, this would be calculated as  $245 - 24 = 221$ . The value 221 is the diameter in pixels.
  - d. Calculate the radius:  $r = diameter / 2 = (y_{bottom} - y_{top}) / 2$ . Using the values in the example above, this is calculated as  $221 / 2 = 110.5$ . The value 110.5 is your radius. This is rounded up to 111, or you can use 110 as the radius for a slightly smaller circle.
2. Find the center coordinates (x,y).
  - The center's y coordinate is the midpoint between  $y_{top}$  and  $y_{bottom}$ .
  - Using the example above, this is calculated as:
$$y = (y_{top} + y_{bottom}) / 2$$
$$(24 + 245) / 2 = 269 / 2 = 134.5$$
  - The y value of 134.5 can be rounded up to 135.

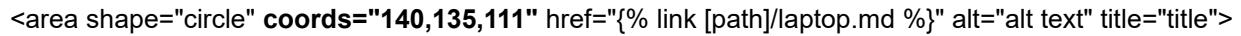
- The center's x coordinate should be approximately the same for the top and bottom edges, as they align vertically in a circle. Since  $x = 139$  (top) and  $x = 140$  (bottom), take the average, or choose one for simplicity:

$x = (139 + 140) / 2 = 279 / 2 = 139.5$ . Round this value up to 140.

- The center coordinate has been calculated as  $(140, 135)$ .

3. Write the `area` tag.

- Combine the center ( $x, y$ ) and radius  $r$  values into the `coords` attribute.
- Using the values in the example above,  $x = 140$ ,  $y = 135$ ,  $r = 111$ . The circle code is written as:

```
<area shape="circle" coords="140,135,111" href="#"> alt="alt text" title="title">
```