Object oriented programming — Part 2

Simon Pfreunschuh

October 14, 2020

1 Understanding `__subclasshook__`

- Use the Python documentation to understand what's the purpose of the code shown below. A good starting point is the official documentation of the `abc` module or the `help` function of the `IPython` interpreter.

```
1  from abc import ABC, abstractmethod
2
3  class DiagramComponent(ABC):
4      ...
5
6      @classmethod
7      def __subclasshook__(cls, C):
8          if cls is DiagramComponent:
9              attributes = set(dir(C))
10             if (set(cls.__abstractmethods__) <= attributes and
11                 set(cls.__abstractproperties__) <= attributes):
12                 return True
13         return NotImplemented
```

- Select a presenter in your breakout room that can explain the functionality implemented in each line.

2 Running the diagram examples

- Clone the code of the `diagrams` example package from `https://github.com/simonpf/diagrams`. The `diagrams` package is an improved implementation of the diagram module we worked with during last lecture. This time it has both an object oriented and a procedural API.

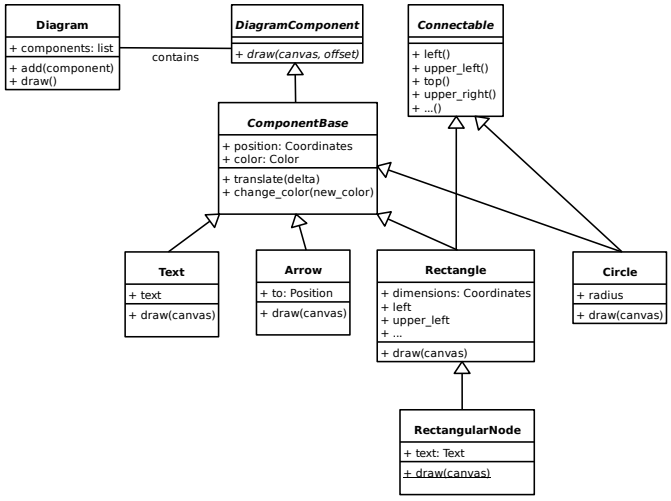

Figure 1: UML diagram of =diagrams= package with the new =Circle= class.

- The `examples` folder contains two examples of how to use the two different APIs in the `procedural.py` and `object_oriented.py` files. Run both example files.
- Open the scripts in an editor and compare the two APIs.

3 Adding a new component to the OO API.

- Implement the `Circle` component class shown in the UML diagram below
- To draw a circle you can use the `create_oval` member method of the `tkinter` canvas:

```
1 canvas.draw_oval(x_1, y_1, x_2, y_2, color="...")
```

  where (`x_1`, `y_1`) and (`x_2`, `y_2`) are the upper left and lower right corners of the bounding box of the circle.
- To test your implementation run the `object_oriented_circle.py` script. If it works it should draw two circles connected by an arrow.

4 Adding a new component to the procedural API.

- Add the functionality to draw a circle also to the procedural API.