Python Automat Cheat Sheet

by amicheletti via cheatography.com/39488/cs/12281/

Basic

`@machine.state()`	Define a state (use `initial=True` to set initial state)
`@machine.input()`	Define input. Body must be empty
`@machine.output()`	Define output. Make processing here

To define transitions:
`state.upon(input, enter=next_state, outputs=[]`

Graphviz

The `MethodicalMachine` class has an function named `asDigraph()`

You can use it to create a `Digraph` object from the `graphviz` package.

Then you can manipulate it as any Digraph from graphviz.
To render it, you can use `Digraph.render(filename)`

For example:

```
g = _machine.asDigraph()
g.render("_machine.gv")
```

Serializing

You can serialize the Machine. To do that, you must define serialized values for each state:

```
@machine.state(serialized="on")
```

Then you define a serializer function which will receive this state serialized value:

```
@machine.serializer()
def save(self, state):
return {"is-it-on": state}
```

Then the unserializer:

```
@machine.unserializer()
def _restore(self, blob):
return blob["is-it-on"]
```