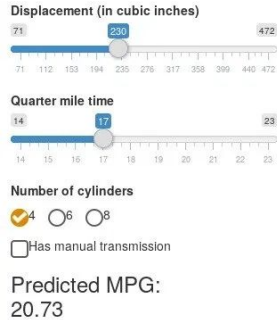


Reactor Modeling Sandbox

Frontend



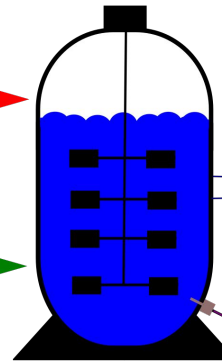
Model inputs \longrightarrow *IndPenSim* \longrightarrow Model outputs

Manual control

- Substrate flow rate (F_s)
- Water for injection (F_w)
- Phenylacetic acid flow rate (F_{PAA})
- Aeration rate (F_g)
- Agitator rate (RPM)
- Discharge rate (F_{dis})
- Water for injection (F_w)
- Nitrogen (F_N)

Automatic control

- Acid/Base flowrate ($F_{a/b}$)
- Cooling water flowrate (F_c)



Off-line measurements

- Penicillin (P)
- Phenylacetic acid (PAA)
- Nitrogen (N)
- Viscosity (μ_{vis})

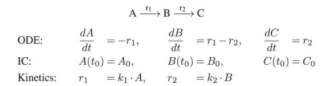
On-line measurements

- Temperature (T)
- pH (pH)
- Dissolved Oxygen (DO_2)
- Weight (W)
- Agitator power (P_{ag})
- Off-gas CO_2 ($CO_{2,og}$)
- Off-gas O_2 ($O_{2,og}$)

PAT measurements

- Raman spectroscopy (*Spec*)

Backend



```
class SequentialKinetic(BioprocessModel):
    def __init__(self, t, y, sw):
        # Unpacks the state vector. The states are alphabetically ordered.
        A, B, C = y
        # Unpacks the model parameters.
        k1 = self.model_parameters["k1"]
        # The "sw" (switches) argument represents a list of booleans,
        # which are true after the corresponding event was hit (False -> True)
        if sw[0]:
            k2 = self.model_parameters["k2"]
        else:
            k2 = 0
        # Defines the derivatives.
        dAdt = -k1*A
        dBdt = k1*A - k2*B
        dCdt = k2*B
        # Returns the derivatives. The order corresponds to the state vector.
        return [dAdt, dBdt, dCdt]
```

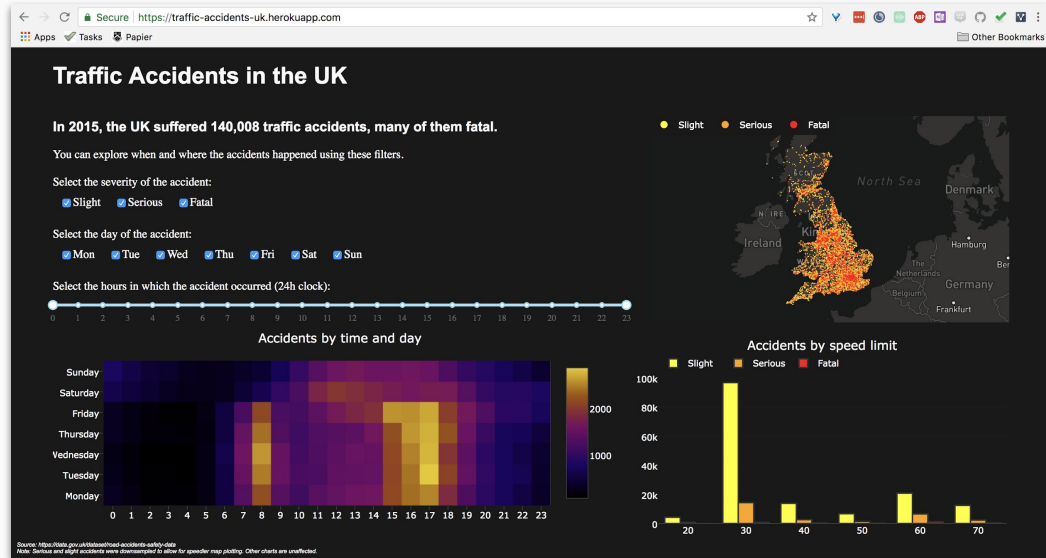
Dashboard/Interface Tech

Tool	Language	Compatibility with Jupyter	Processing Speed	Customizability
Bokeh	Python, JavaScript	Natively renders visualizations	Low	Medium
Dash	Python, HTML, CSS, JavaScript	Supported by plotly extension	Medium	High
Voila	Python	Natively renders visualizations	Medium	Medium
Streamlit	Python	Not compatible	High	Low

We chose Dash!

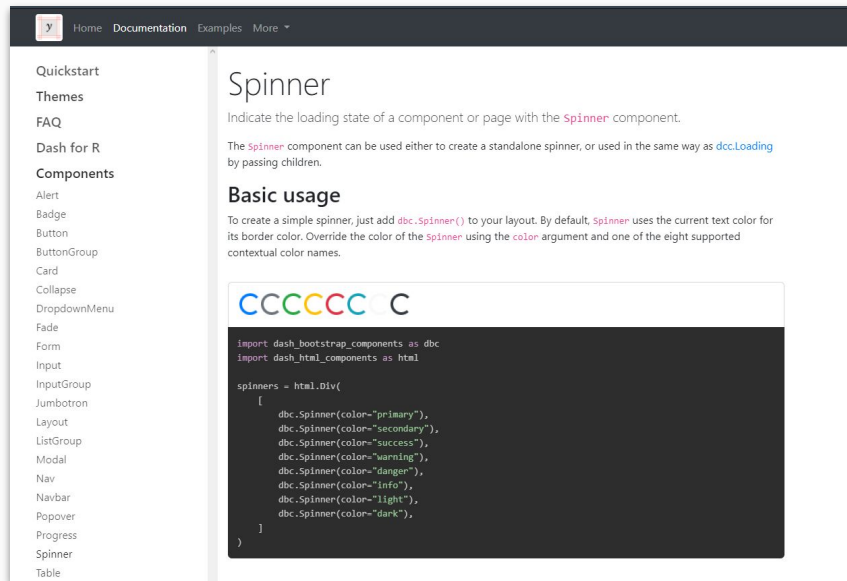
What is Dash?

- A Python framework for building interactive web apps.
- Apps are rendered in the web browser.
- Deploys apps to servers and shares them through URLs.



Dash: Advantages

- Customizability
- Large community examples
- Many components



Dash: Drawbacks

- HTML and CSS heavy
- More lines of code
- Not as easy to deploy as Streamlit

```
import dash_html_components as html

html.Div([
    html.H1('Hello Dash'),
    html.Div([
        html.P('Dash converts Python classes into HTML'),
        html.P("This conversion happens behind the scenes by Dash's JavaScript front-end")
    ])
])
```

which gets converted (behind the scenes) into the following HTML in your web-app:

```
<div>
  <h1>Hello Dash</h1>
  <div>
    <p>Dash converts Python classes into HTML</p>
    <p>This conversion happens behind the scenes by Dash's JavaScript front-end</p>
  </div>
</div>
```