

Marrying form and function: design takeaways from an R package for estimating real-time epidemic trends from multiple streams of surveillance data

Saras Windecker

Infectious Disease Ecology & Modelling team
The Kids Research Institute Australia



**I acknowledge the Traditional Owners of the unceded
sovereign land of the Turrbal and Jagera peoples, on
whose land I live and work. I pay my respects to their
Elders past, present, and future.**

COVID-19 emphasised the importance of monitoring surveillance timeseries to **estimate infection trends in near-real time**.

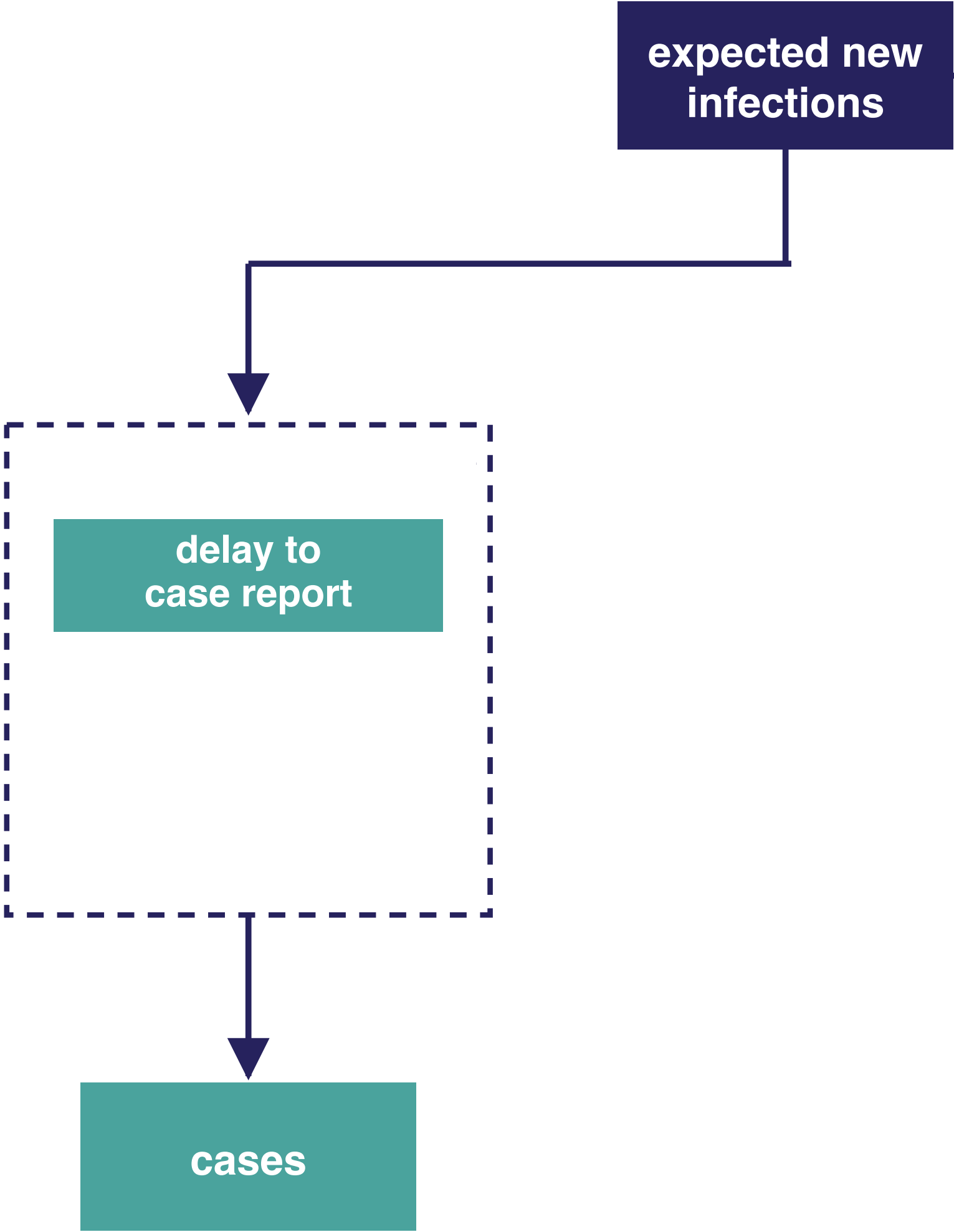


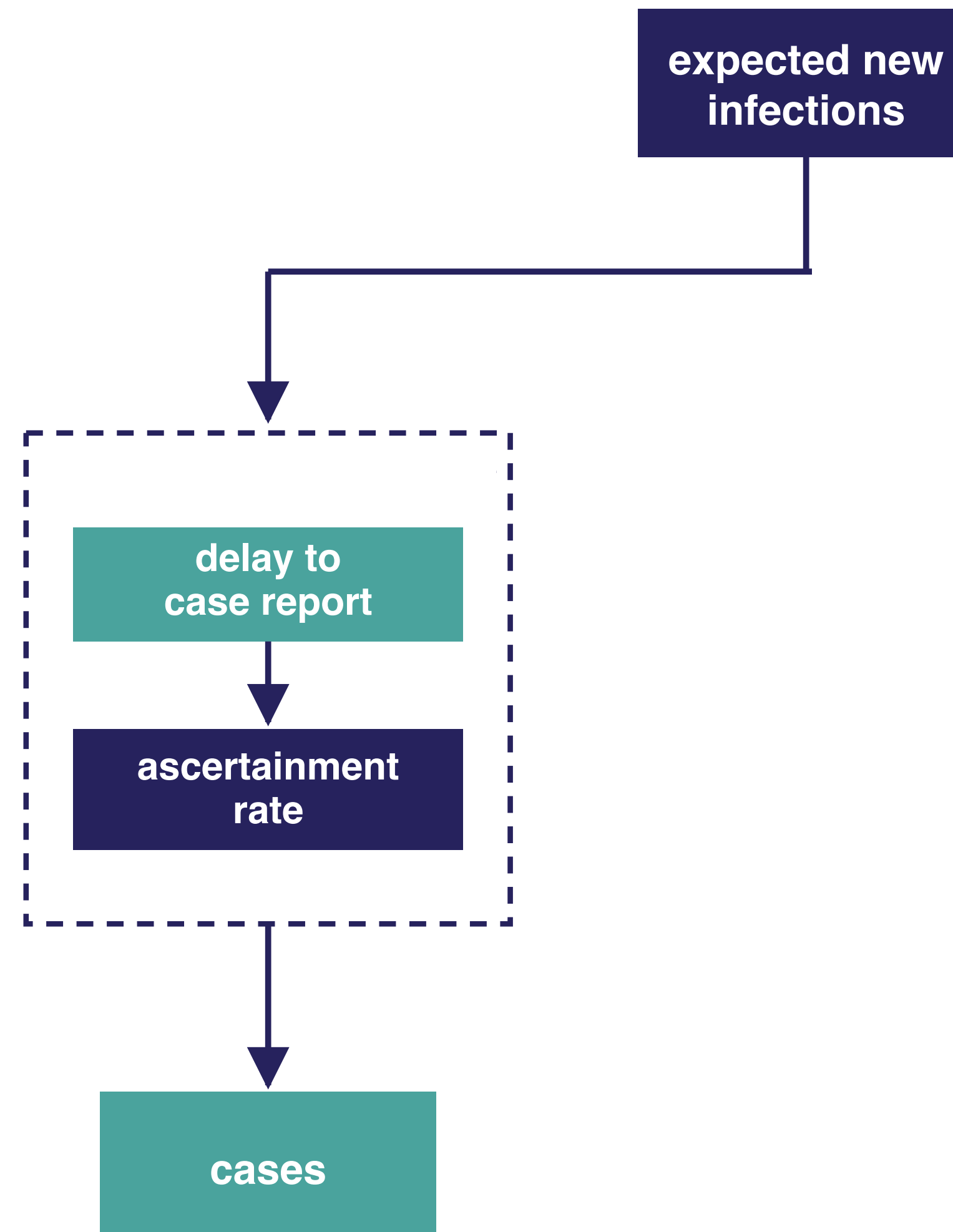
Infection
inference
model

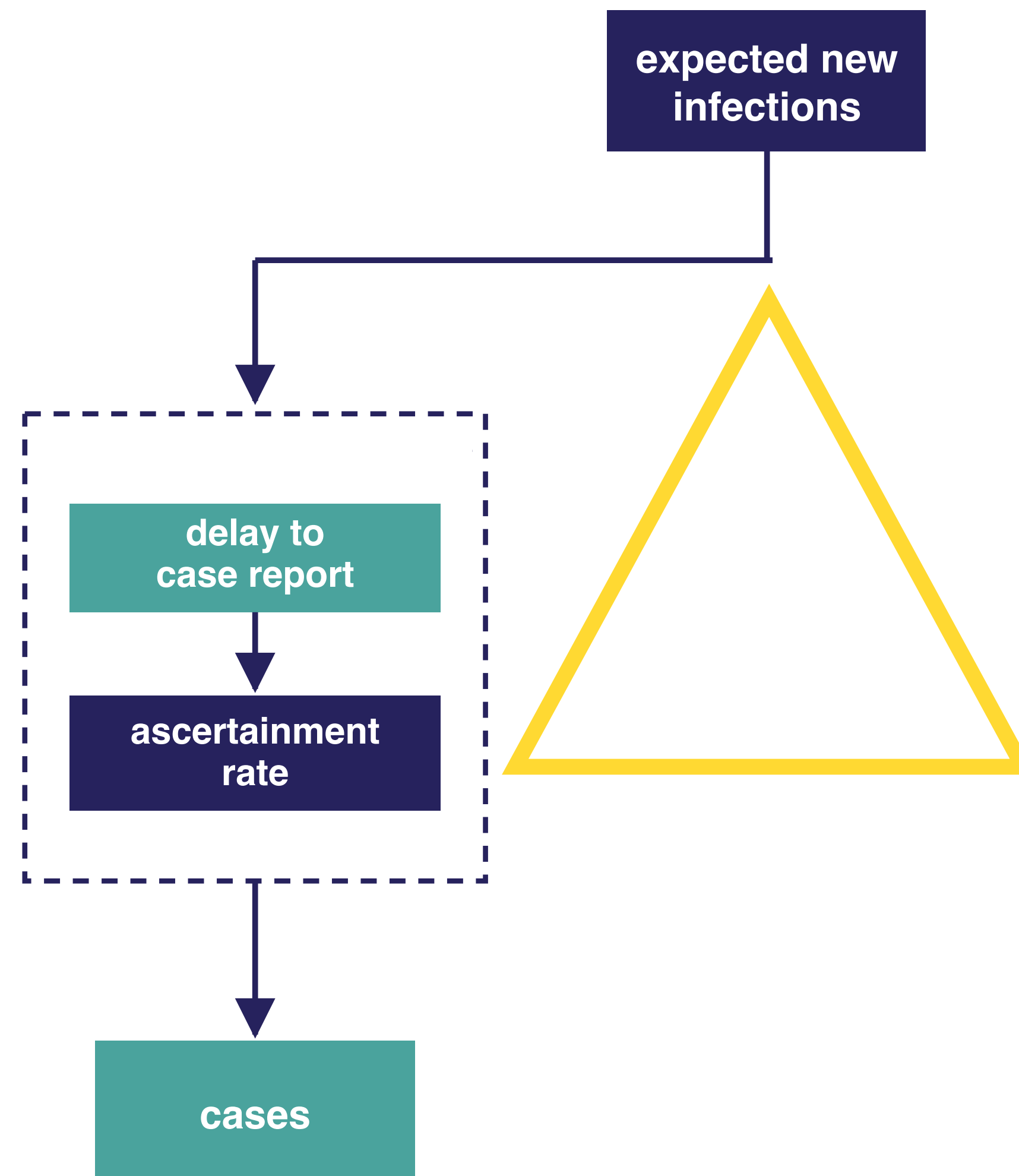


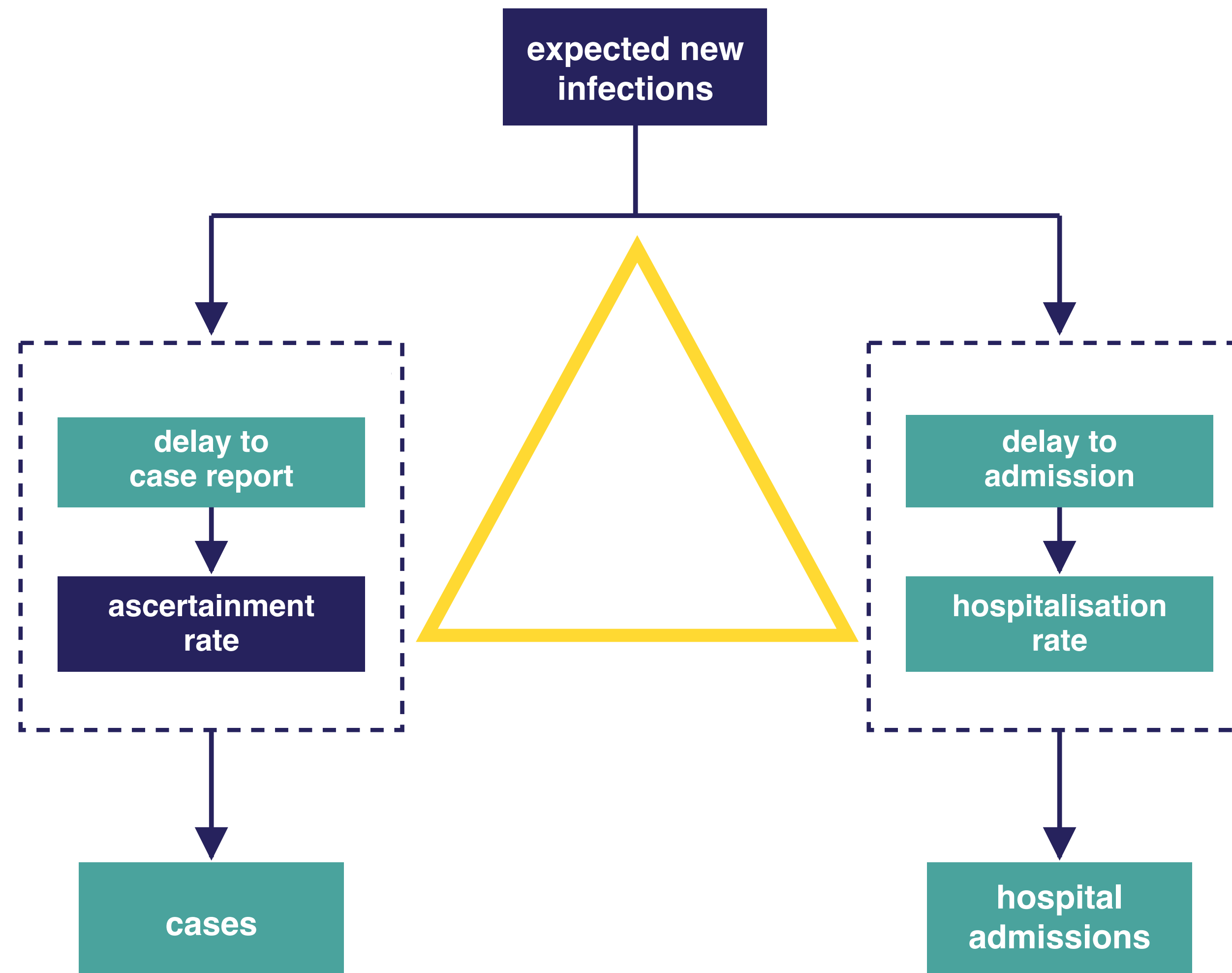
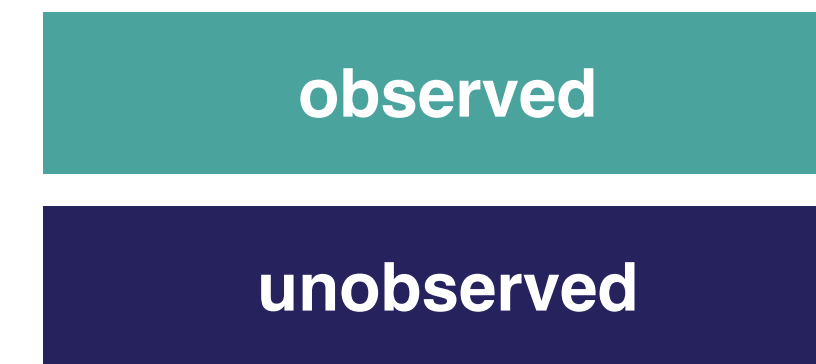
COVID-19 emphasised the importance of monitoring surveillance timeseries to **estimate infection trends in near-real time**.

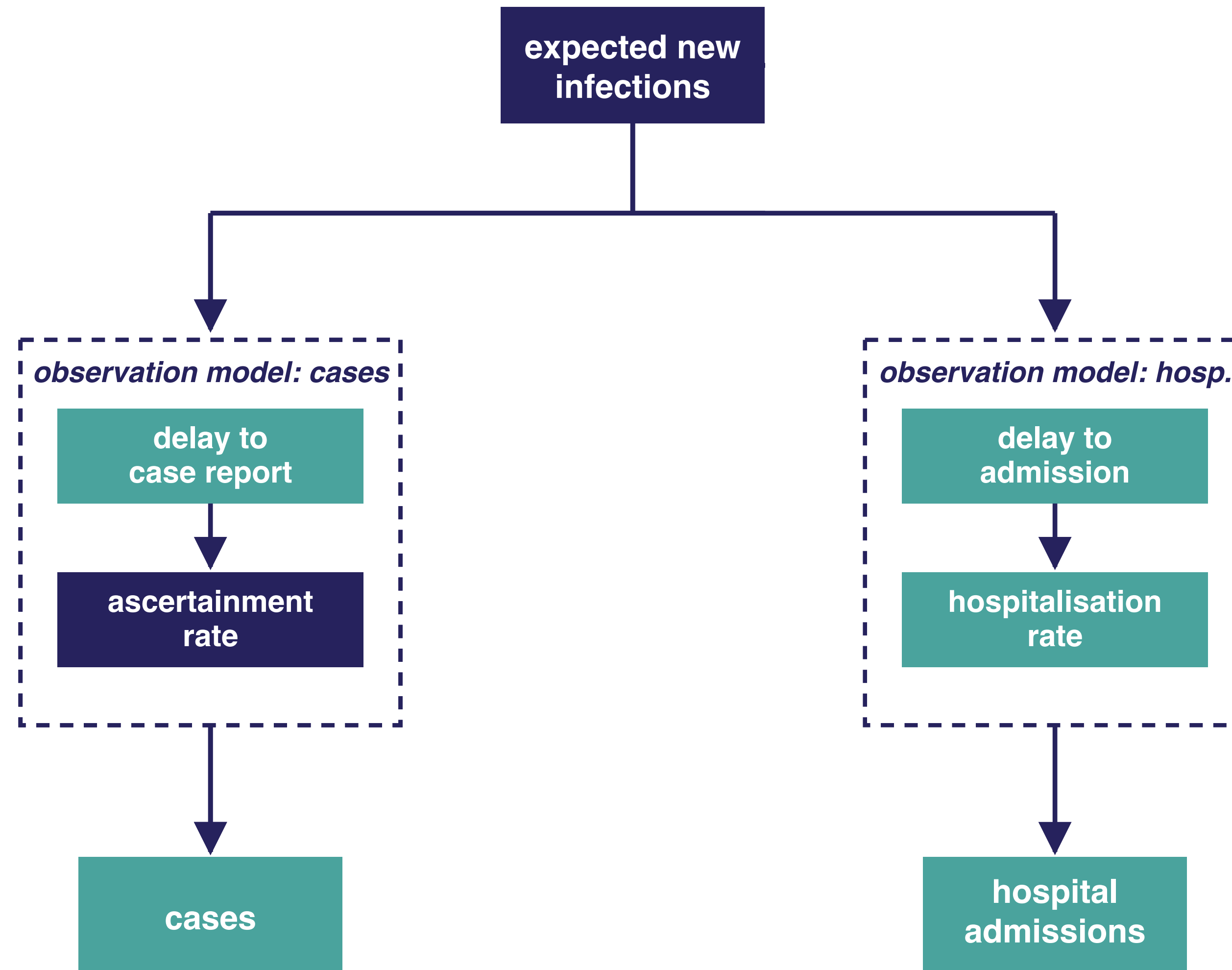


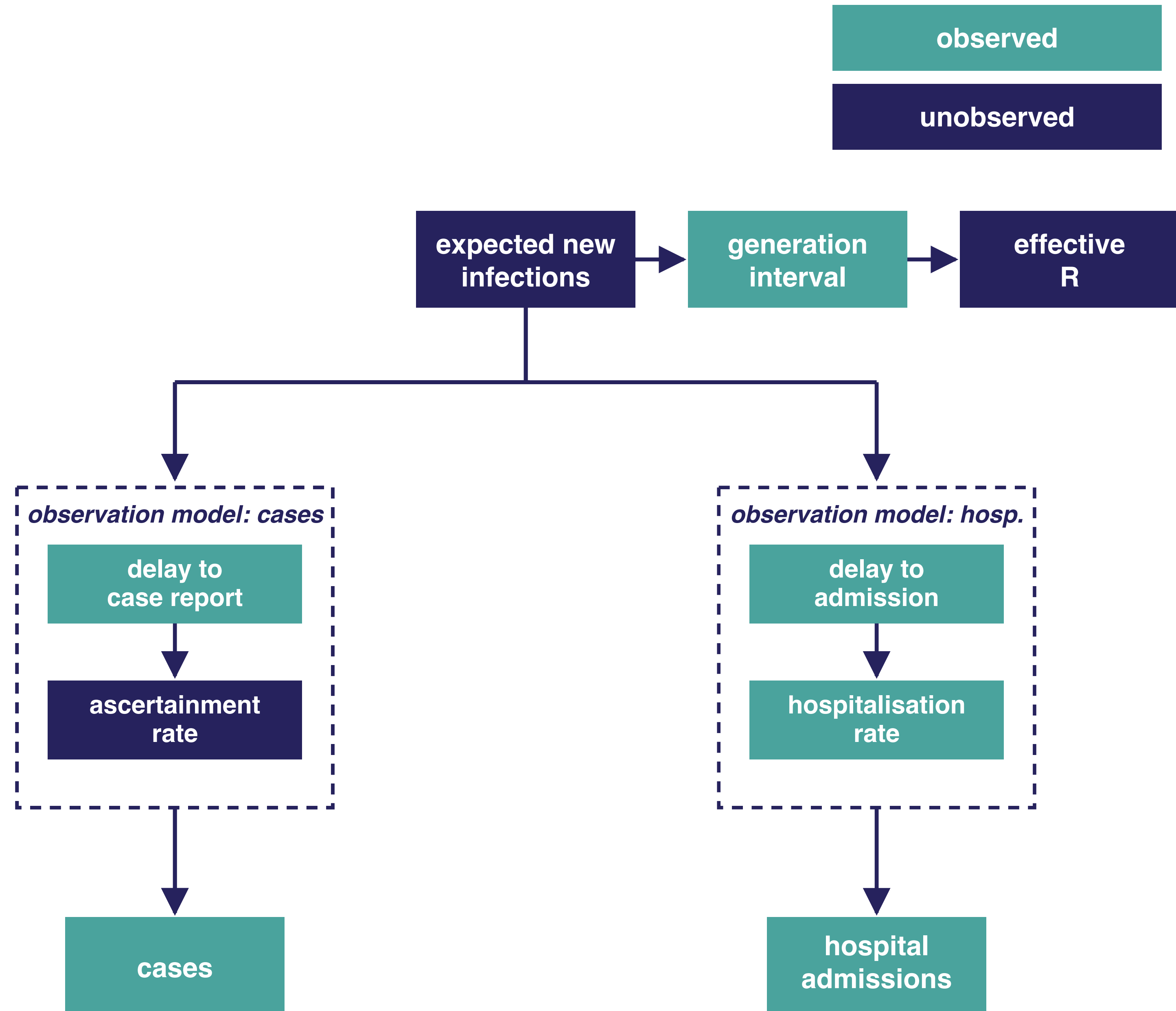


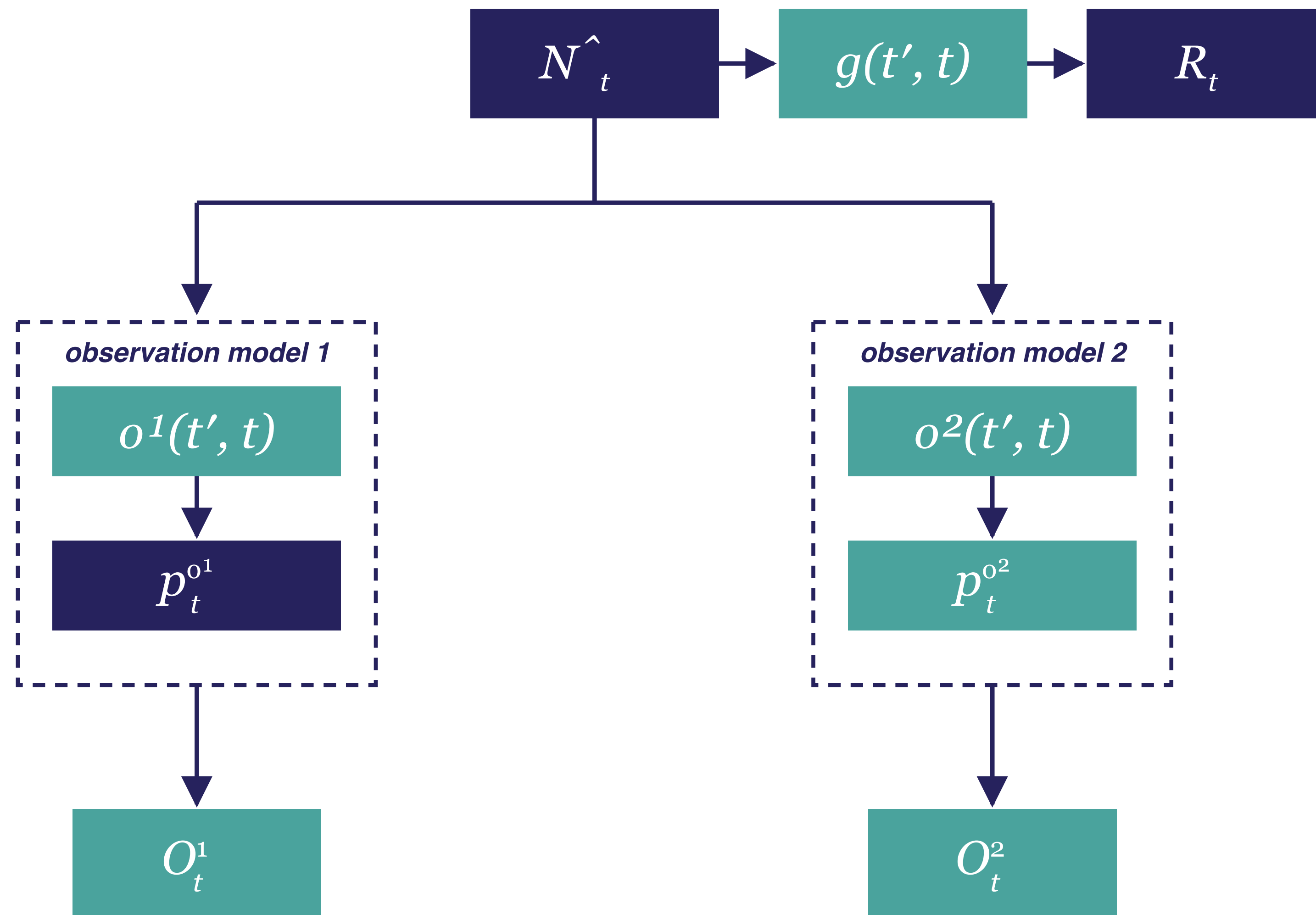




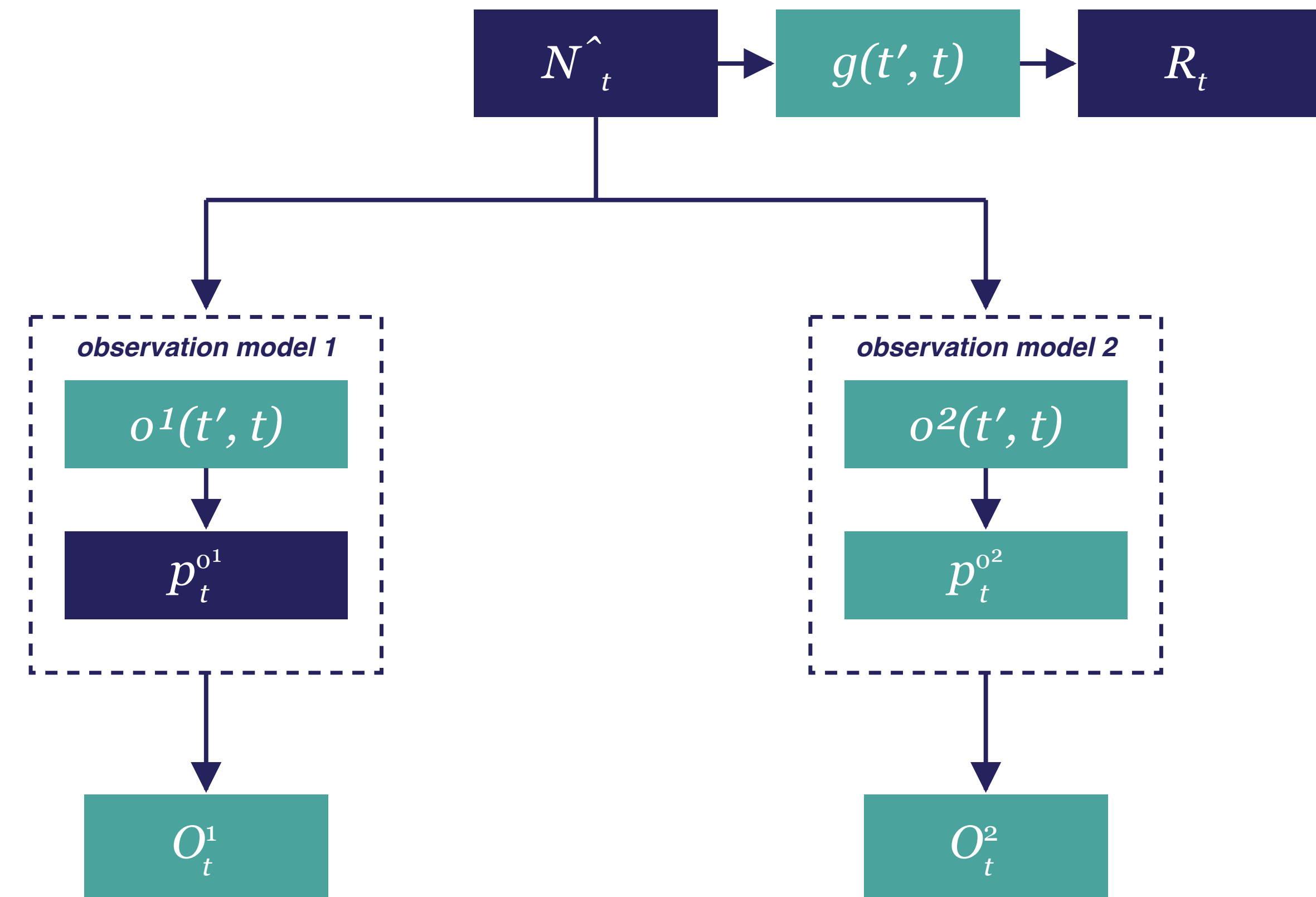








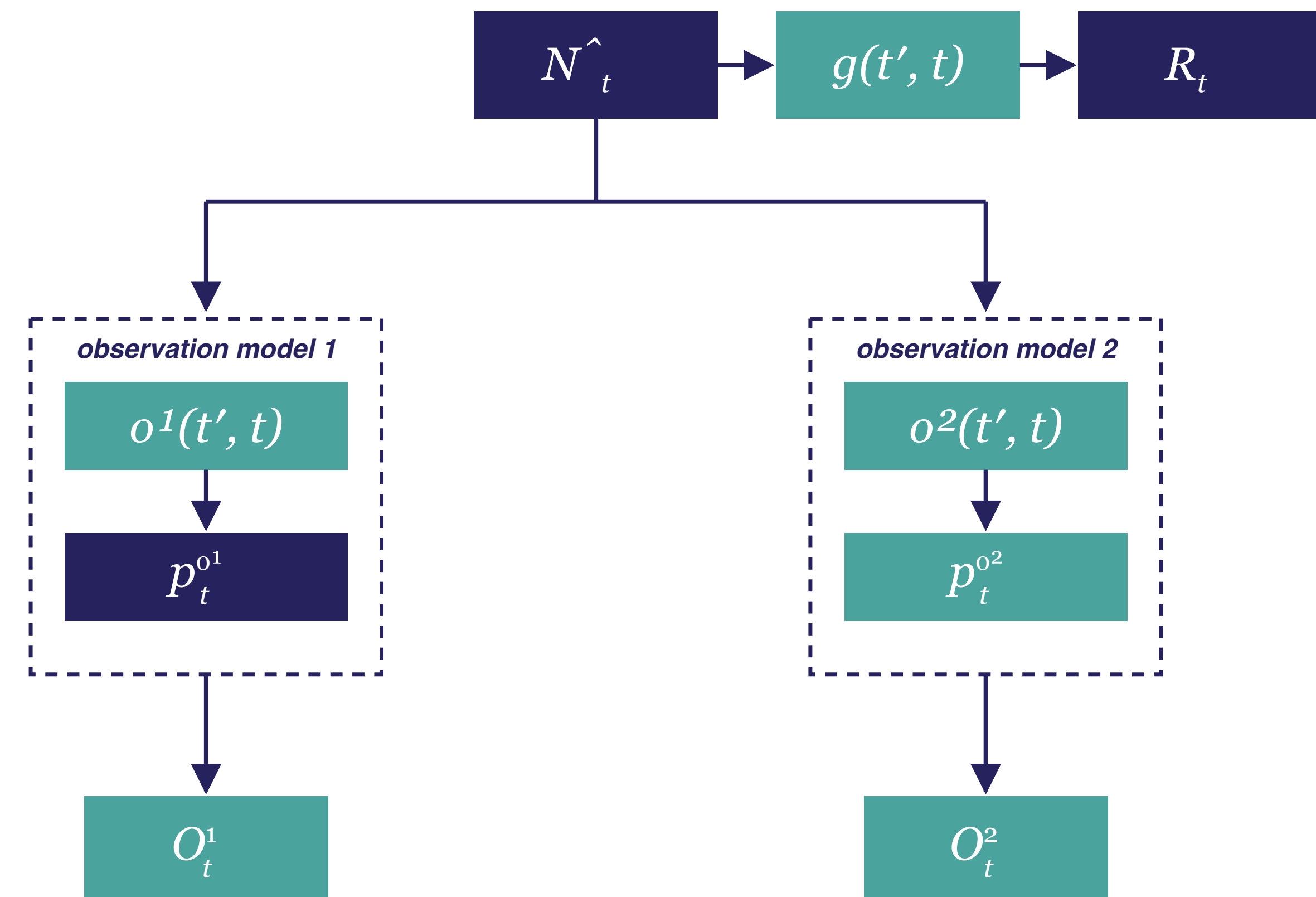
Key design feature: stream-agnostic design



Key design feature: stream-agnostic design

The observation model constructor functions are agnostic of type.

Because we won't know what data will be available.



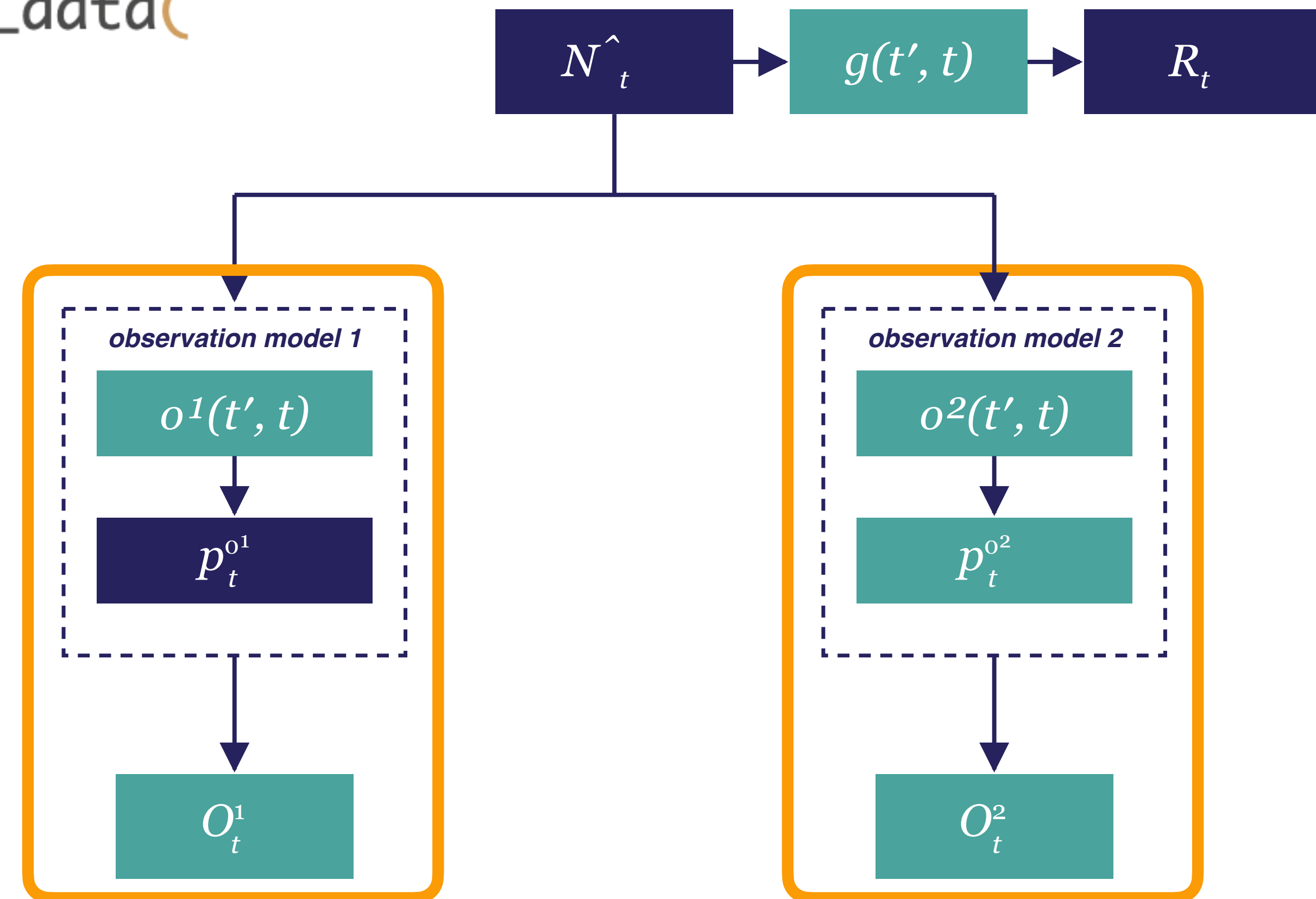
Key design feature: stream-agnostic design

```
hospitalisations = define_observation_data(
```

```
)
```

```
cases = define_observation_data(
```

```
)
```



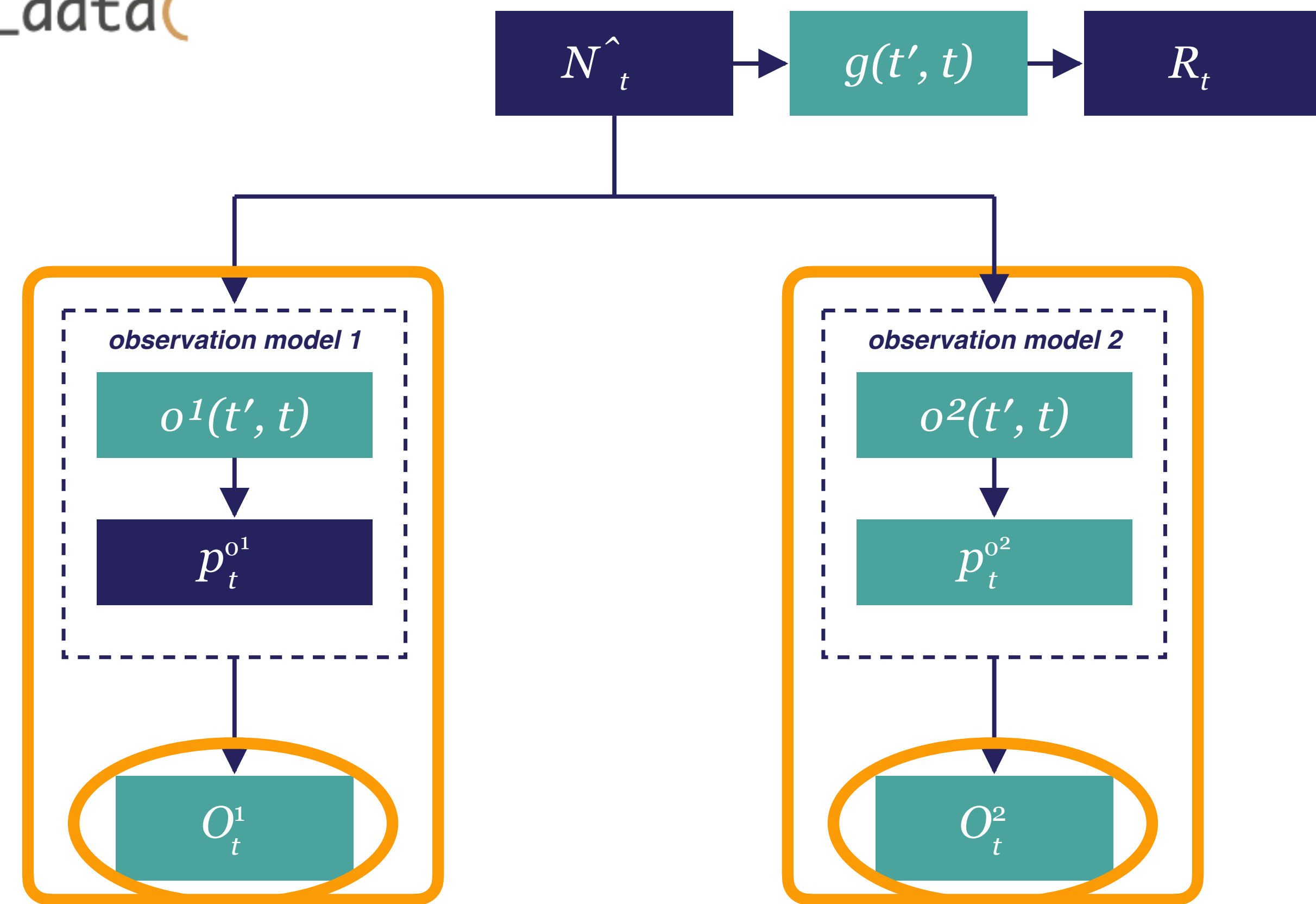
Key design feature: stream-agnostic design

```
hospitalisations = define_observation_data(  
    timeseries_data = hosp_timeseries,
```

)

```
cases = define_observation_data(  
    timeseries_data = case_timeseries,
```

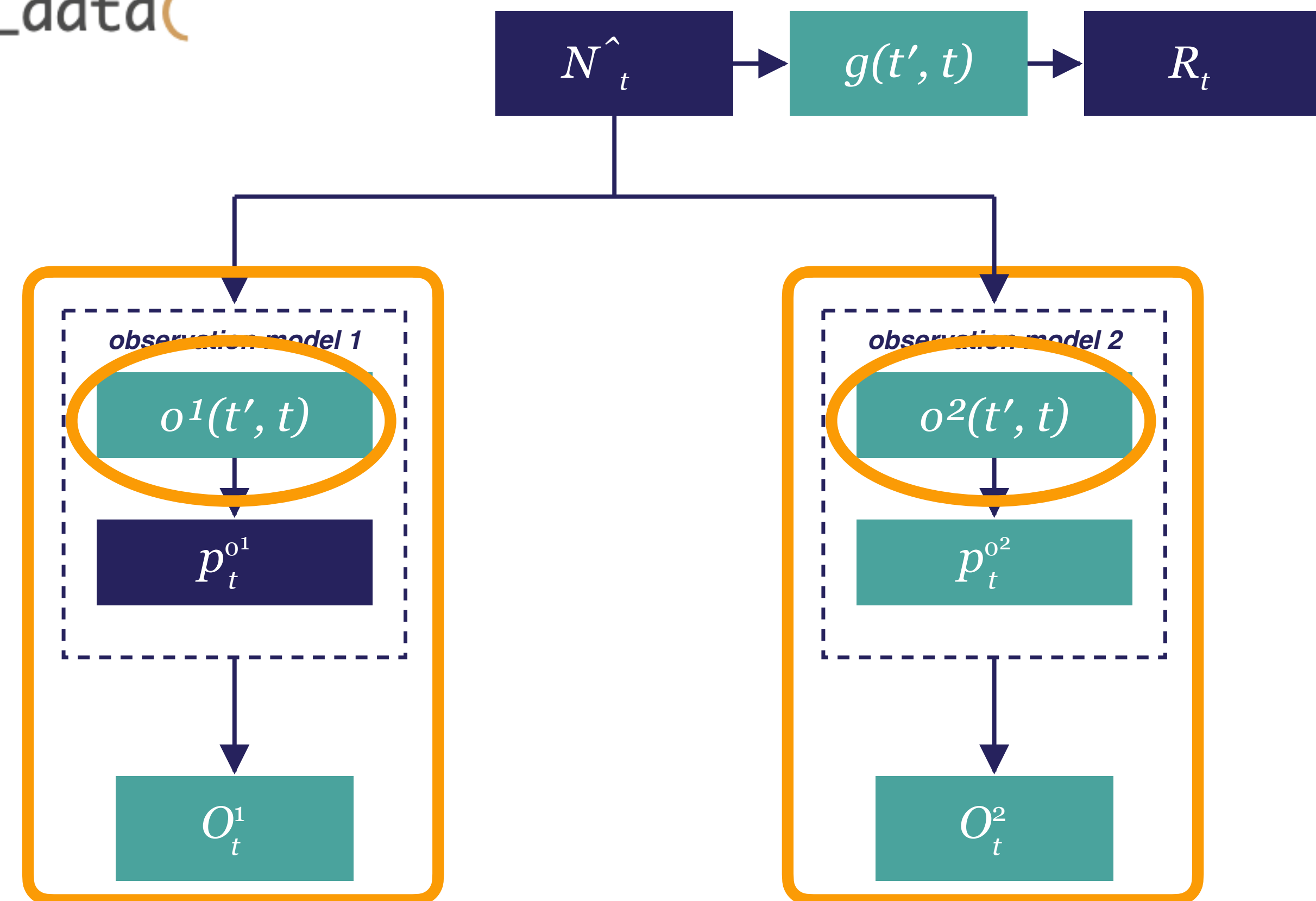
)



Key design feature: stream-agnostic design

```
hospitalisations = define_observation_data(  
    timeseries_data = hosp_timeseries,  
    delay_from_infection = hosp_delay,  
    )
```

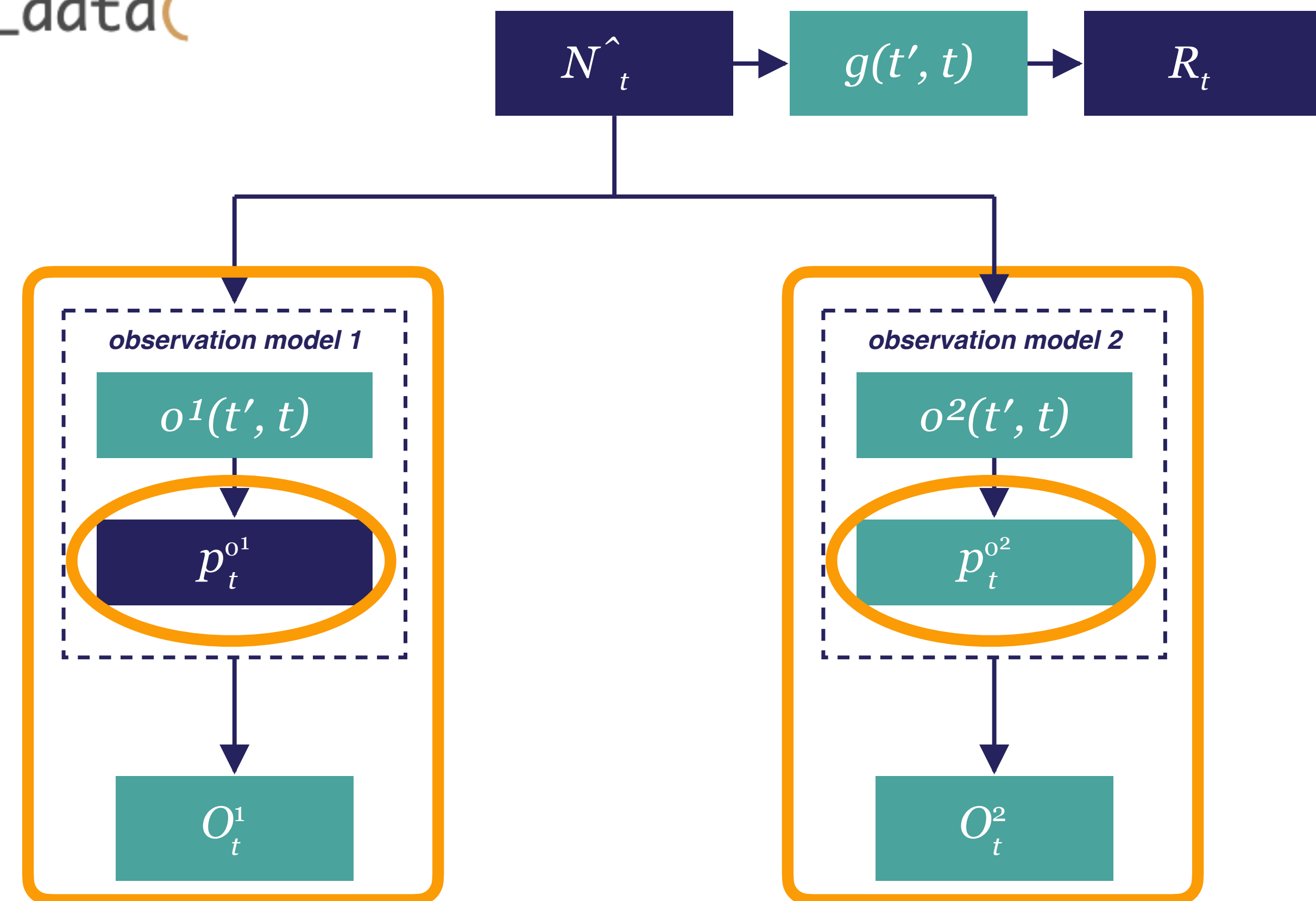
```
cases = define_observation_data(  
    timeseries_data = case_timeseries,  
    delay_from_infection = case_delay,  
    )
```



Key design feature: stream-agnostic design

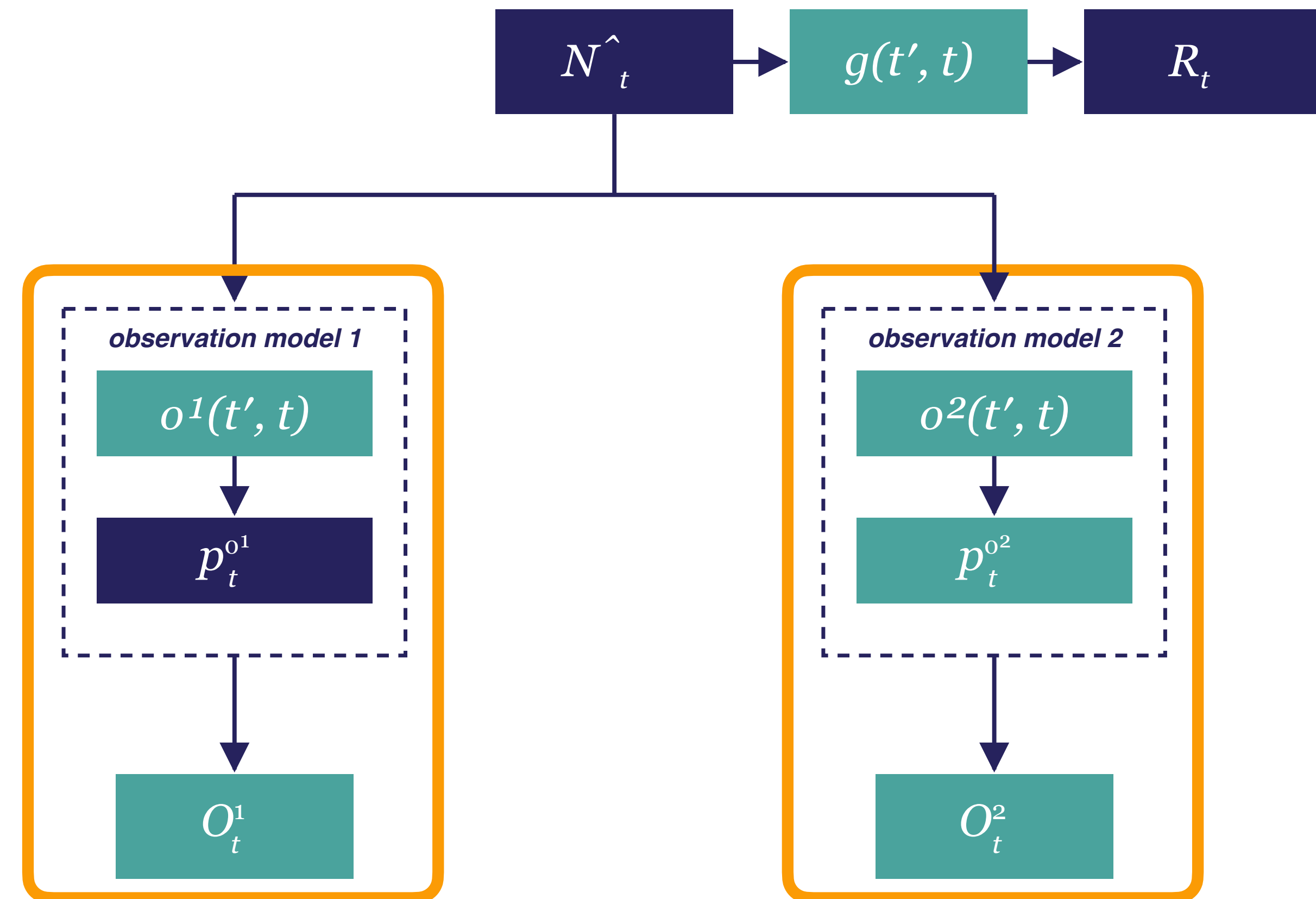
```
hospitalisations = define_observation_data(  
    timeseries_data = hosp_timeseries,  
    delay_from_infection = hosp_delay,  
    proportion_infections = ihr  
)
```

```
cases = define_observation_data(  
    timeseries_data = case_timeseries,  
    delay_from_infection = case_delay,  
    proportion_infections = car  
)
```



Key design feature: stream-agnostic design

This parallel design means we can incorporate new data that becomes available without totally reworking the model.

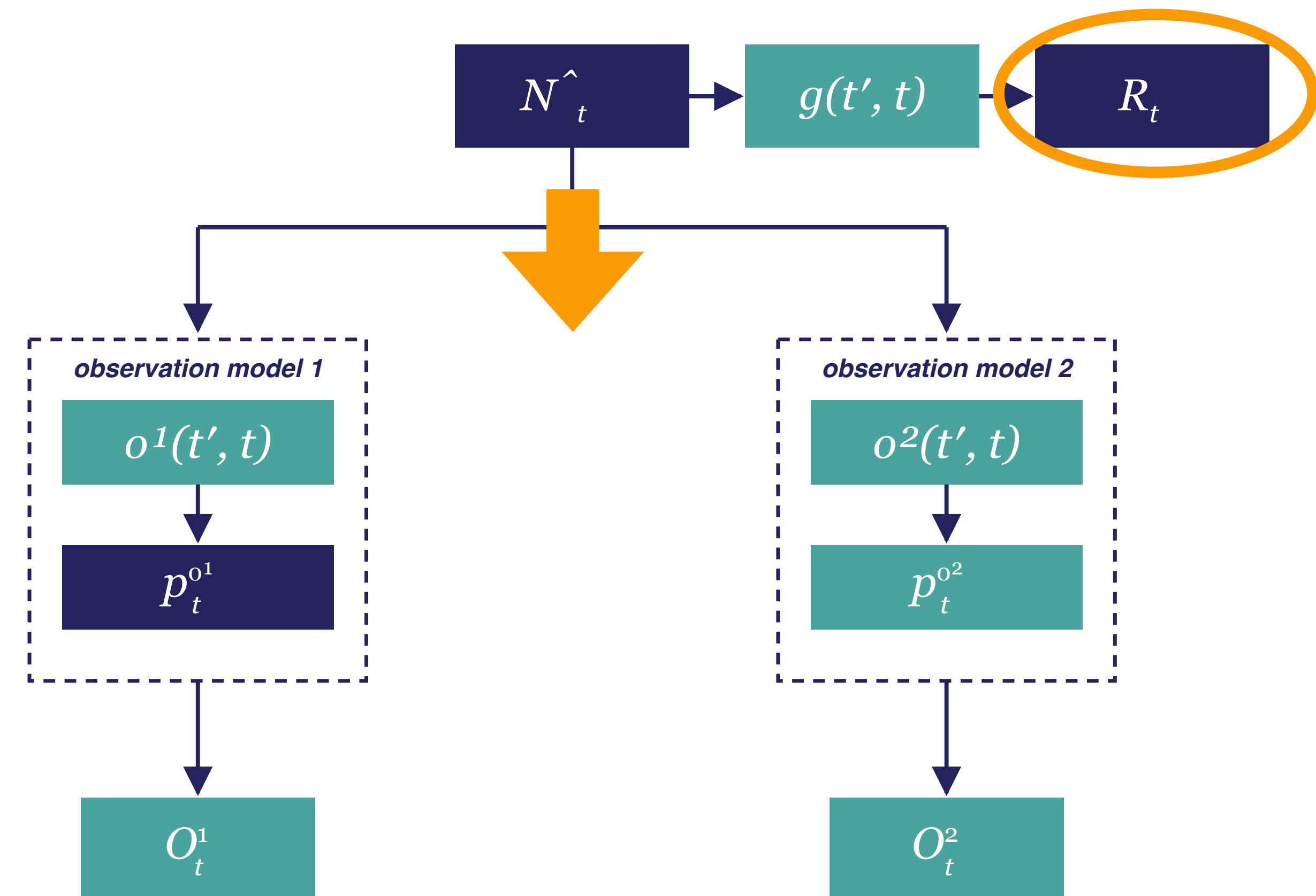


Key design feature: stream-agnostic design

This parallel design means we can incorporate new data that becomes available without totally reworking the model.

Many other design choices that reflect the model.

Aids the user in understanding underlying.



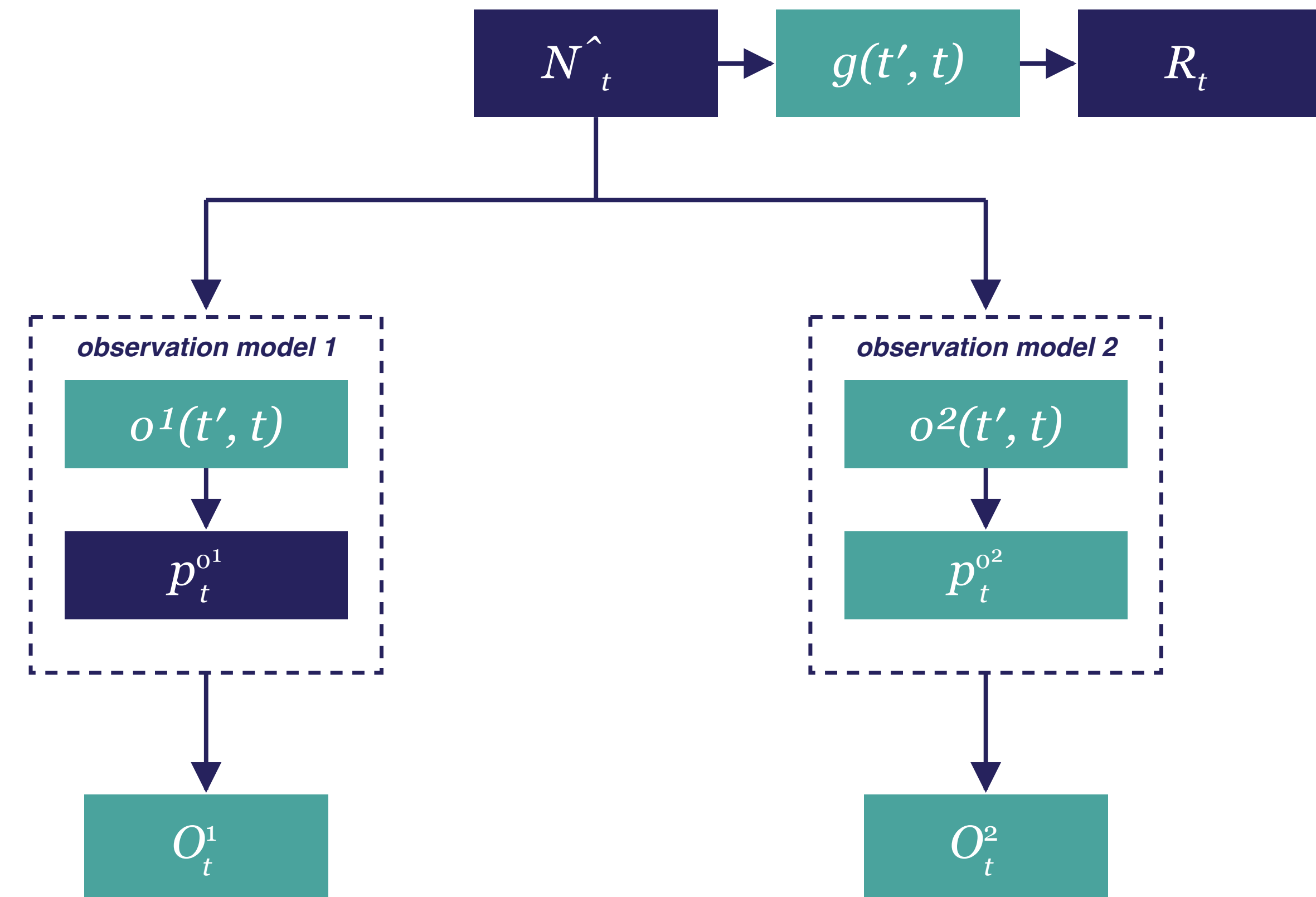
Key design feature: stream-agnostic design

This parallel design means we can incorporate new data that becomes available without totally reworking the model.

Many other design choices that reflect the model.

Aids the user in understanding underlying.

Focus on **flexibility but specificity**.



Thank you!

saras.windecker@thekids.org.au

A WIP: <https://github.com/idem-lab/epiwave>

Some fun other finished work: <https://github.com/acefa-hubs/EpiStrainDynamics>

August Hao

Kate Senior

Nick Golding

