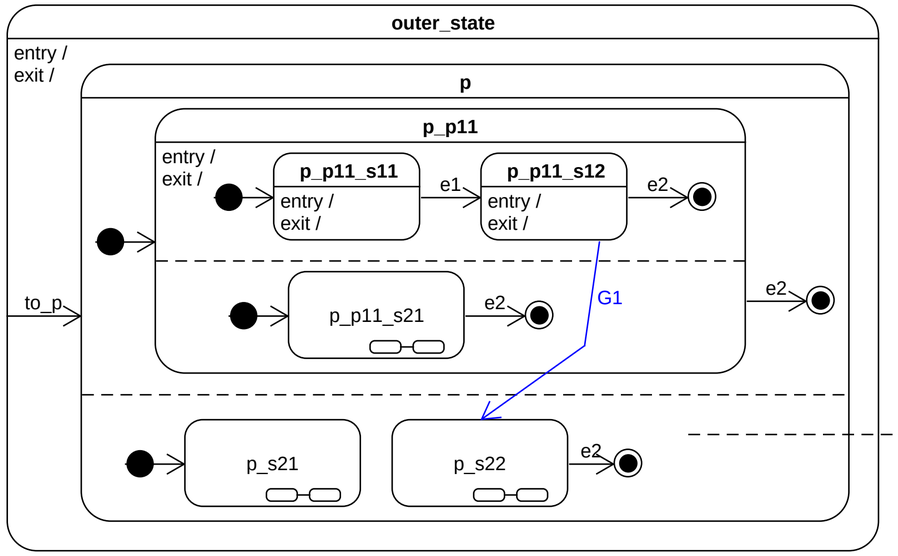


It is p_p11 has cause transitions in p_p11_r1 and p_p11_r2 to their under_hidden_regions, but this happens with an exit

It is p that has cause to cause transitions in p_r1 to under_hidden_region, but this happens with an exit

It is p that has cause to cause a transition in p_r2 to p_s21 via its INIT_SIGNAL

Idea:
* try making an onion from p_s21 to p_p11_s12
* reverse it
* for each item in the onion assign a META_EXIT or META_INIT
* An injector receiving a META_EXIT will only inject an exit_region if the function is in the regions it is injecting to
* If receiving a META_INIT, use peak_meta and see if the function is in the region, if so post_lifo force_region_init

Introducting the second WTF event

Harel Diagram

Mapping onto an orthogonal components

```
payload = namedtuple(
  'META_SIGNAL_PAYLOAD',
  [
    'event',
    'state',
    'previous_state',
    'previous_event'
  ]
)
```

«injectors»

XmlChart

**outer_state**

**p**

```
enter /
  # ...
  _e, _state = self.peel_meta(e)
  if _state:
    self.regions['p']._post_lifo(_e)
  r.regions['p_p11'].post_lifo(
    signal=signals.enter_region)
```

to_p

**p_r1_under_hidden_region**

**p_r1_region**

**p_r1_over_hidden_region**

**p_p11**

```
entry as e/
  _e, _state = self.peel_meta(e)
  if _state:
    self.regions['p']._post_lifo(_e)
  r.regions['p_p11'].post_lifo(
    signal=signals.enter_region)
```

force_region_init

enter_region

exit_region

**p_p11_r1_under_hidden_region**

**p_p11_r1_region**

**p_p11_r1_over_hidden_region**

**p_p11_s11**

entry /
exity /

e1

e2

**p_p11_s12**

```
G1 as e /
  _e = rr.outmost.meta_trans(
    t=p_s21,
    s = p_p11_s12, sig=e)
  rr.outer.post_fifo(_e)
```

force_region_init

enter_region

exit_region

1

EXIT_META_SIGNAL <function p at ...>

**p_r2_under_hidden_region**

**p_r2_region**

**p_r2_over_hidden_region**

**p_s21**

```
entry as e/
  _e, _state = self.peel_meta(e)
  if _state:
    self.regions['p']._post_lifo(_e)
  r.regions['p_p11'].post_lifo(
    signal=signals.enter_region)
```

force_region_init

enter_region

exit_region