Faster Coroutine Pipelines: A Reconstruction

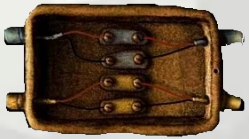
Ruben Pieters

Tom Schrijvers

KU LEUVEN

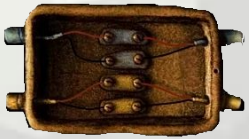
Stream Processing

Stream Processing

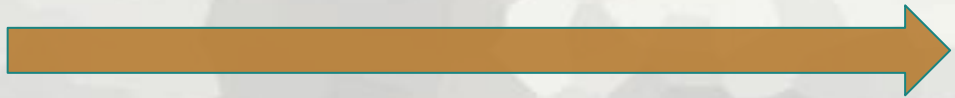


Data

Stream Processing

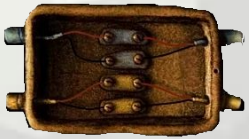


Data



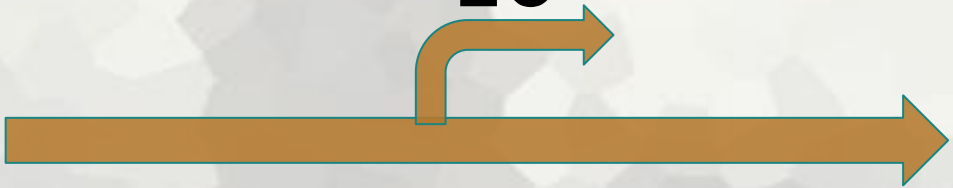
Result

Stream Processing



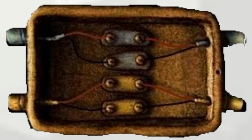
Data

IO



Result

Stream Processing

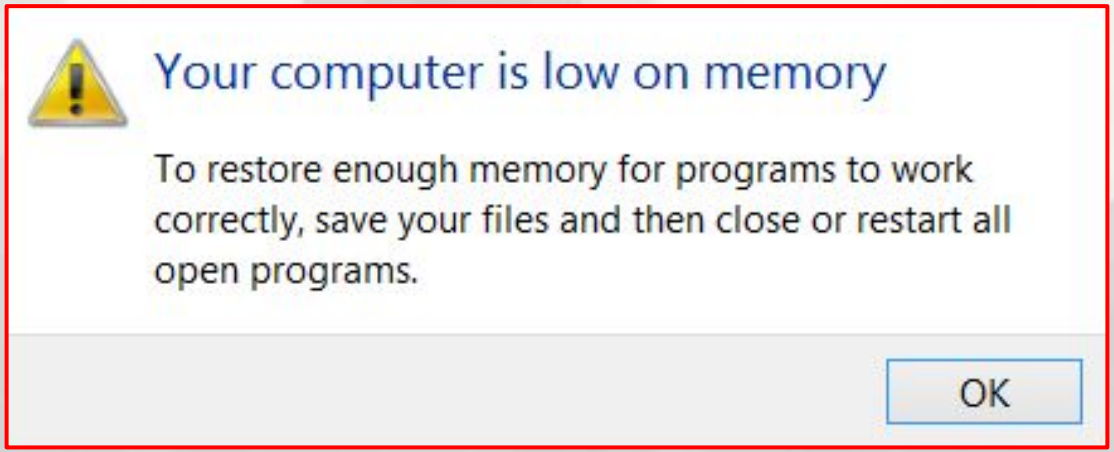


```
[Data] -> IO Result
```

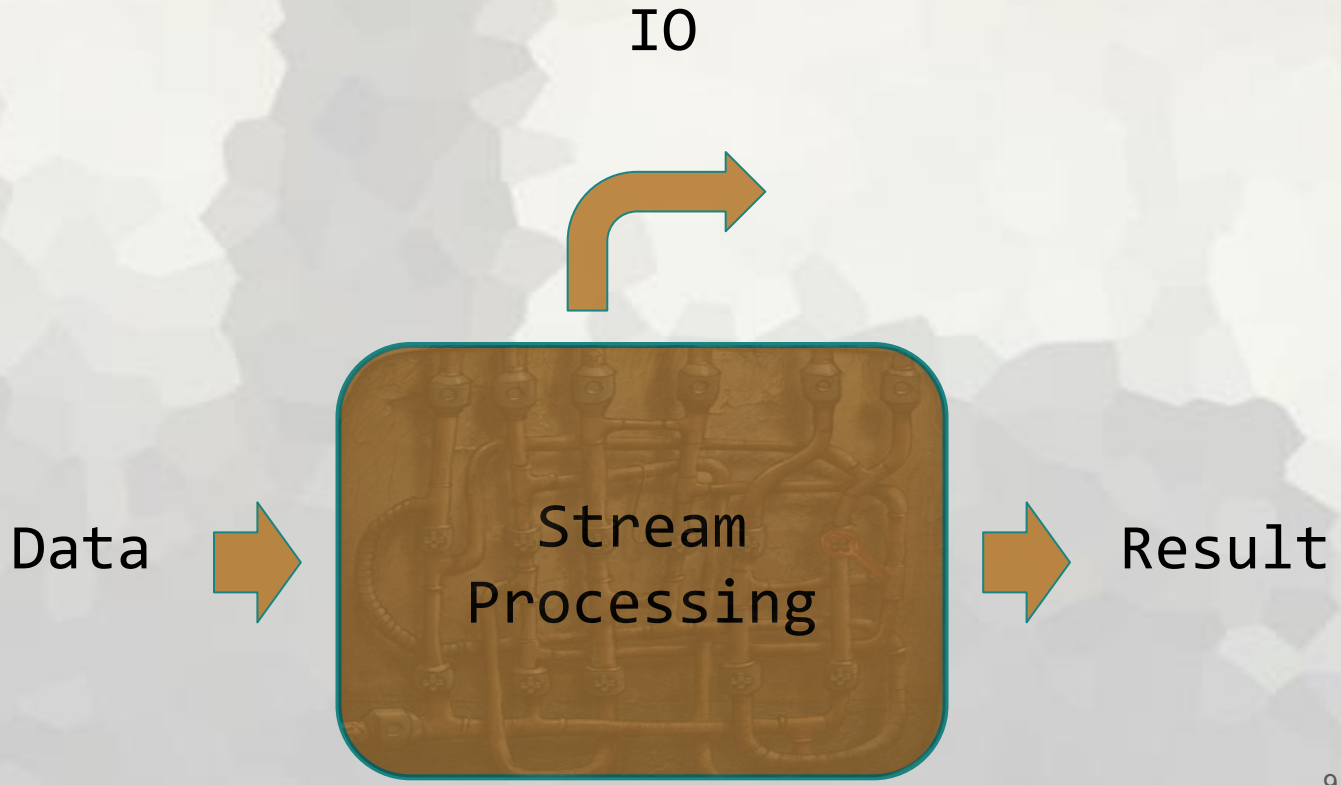
Stream Processing

```
[Data] -> IO Result
```

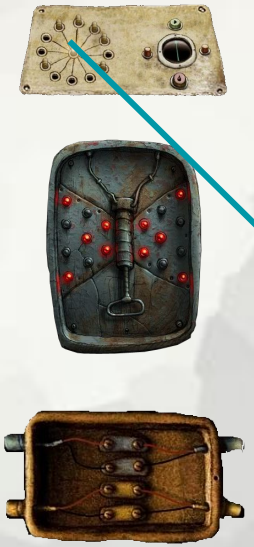
Stream Processing



Stream Processing



Stream Processing



Data

IO

Stream Processing

Result

Stream Processing

Data

IO

Stream Processing

Result

Stream Processing

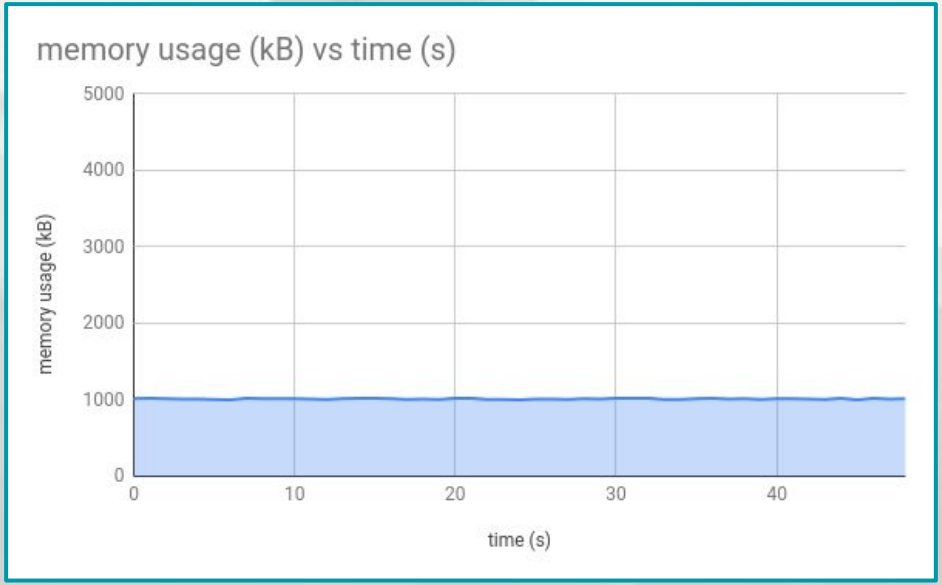
Data

IO

Stream Processing

Result

Stream Processing



Stream Processing

Stream Processing

IO

Data

Result

Stream Processing

Traditional[1,2]

⟹

Alternative[3,4]

[1] *Pipes*, Gonzalez, Haskell Library
[2] *Conduit*, Snoyman, Haskell Library
[3] *Continuations and Transducer Composition*, Shivers and Might, PLDI ‘06
[4] *Faster Coroutine Pipelines*, Spivey, ICFP ‘17

Overview

Goal

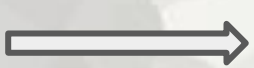
Introduction to

- Traditional Representation[1,2]
- Alternative Representation[3,4]

[1] *Pipes*, Gonzalez, Haskell Library
[2] *Conduit*, Snoyman, Haskell Library
[3] *Continuations and Transducer Composition*, Shivers and Might, PLDI ‘06
[4] *Faster Coroutine Pipelines*, Spivey, ICFP ‘17

Our Paper

Traditional[1,2]

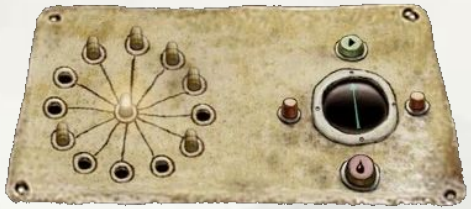


Systematic Derivation

Alternative[3,4]

[1] *Pipes*, Gonzalez, Haskell Library
[2] *Conduit*, Snoyman, Haskell Library
[3] *Continuations and Transducer Composition*, Shivers and Might, PLDI '06
[4] *Faster Coroutine Pipelines*, Spivey, ICFP '17

Story



helloooooooo world!!!

im a nice person!

I only just worked out that the Moon landings were a hoax

helloooooooo world!!!

Step 1: Generate And Tweet

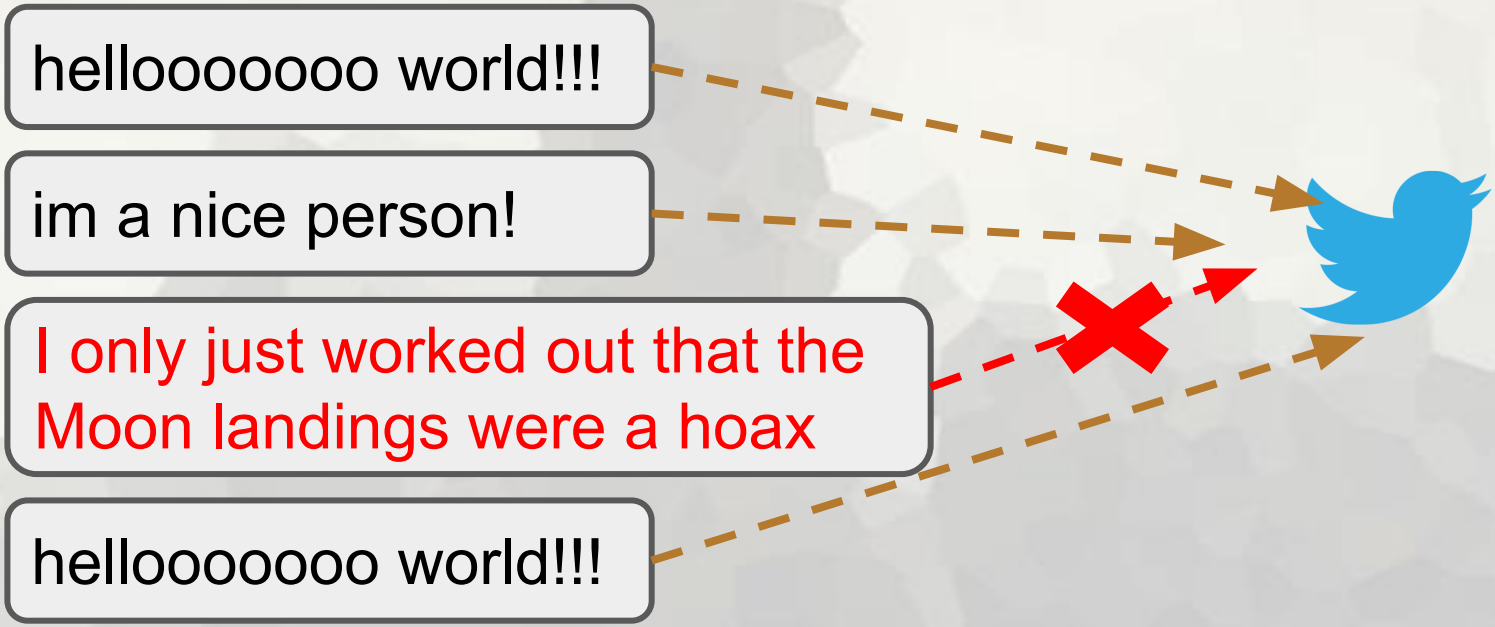
helloooooooo world!!!

im a nice person!

I only just worked out that the Moon landings were a hoax

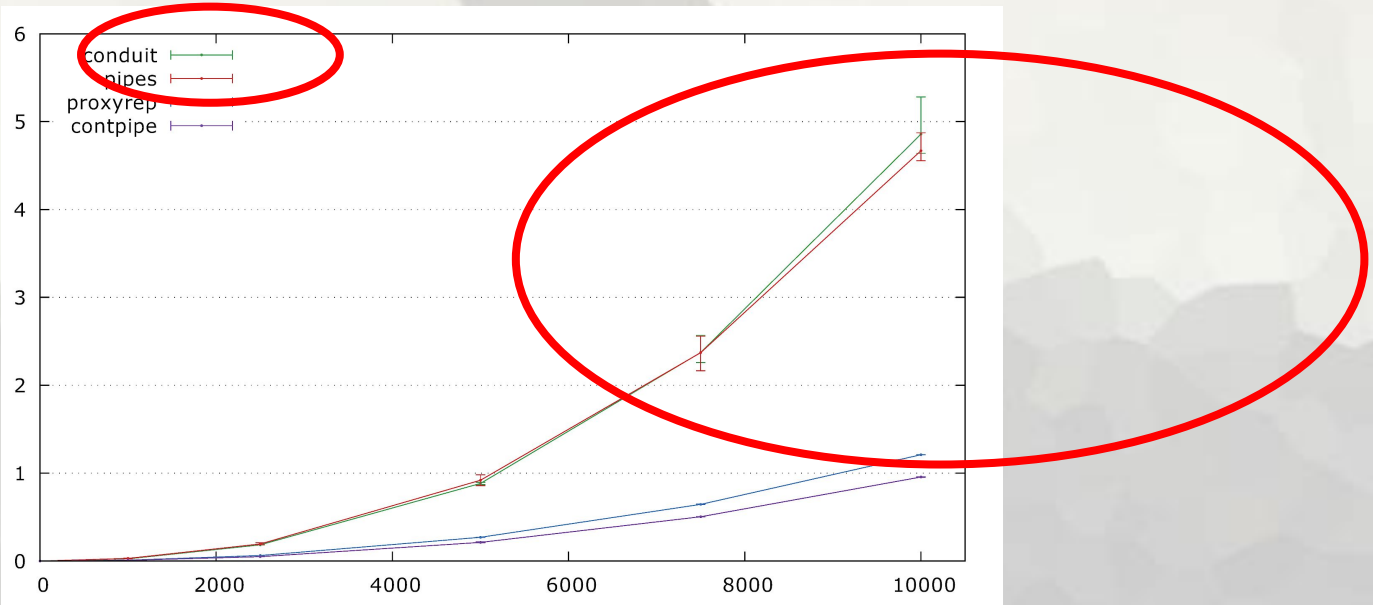
helloooooooo world!!!

Step 2: Filter Bad



Step 3: Filter Duplicates

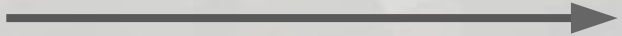
Problem While Filtering Duplicates



Step 1: Generate And Tweet

Producer/Consumer

Pr(oducer) o m r

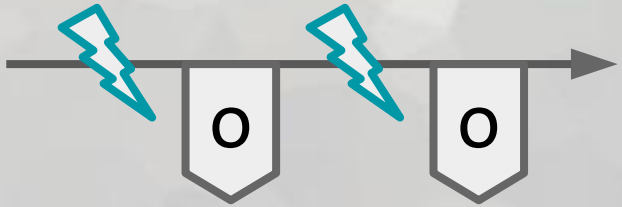


Producer/Consumer

Pr(oducer) o m r

Producer/Consumer

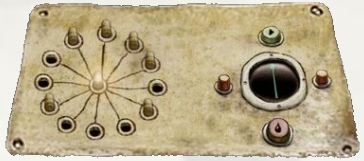
Pr(oducer) o m r



Producer/Consumer

Pr(oducer) o m r

Producer/Consumer

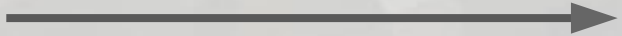


```
: Pr String IO Ø
```

=

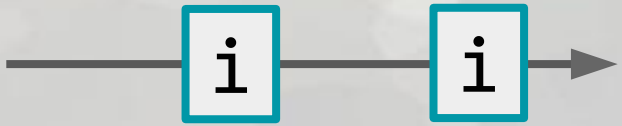
Producer/Consumer

Co(nsumer) i m r



Producer/Consumer

```
Co(nsumer) i m r
```



Producer/Consumer

```
Co(nsumer) i m r
```

Producer/Consumer

Co(nsumer) i m r

Producer/Consumer

```
: Co String IO Ø
```

=

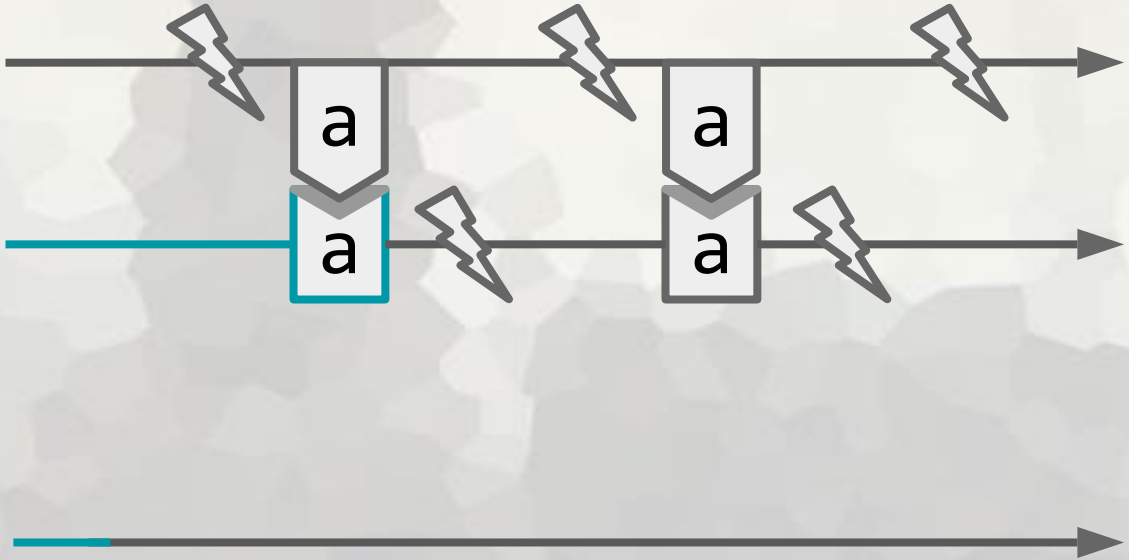
Merge

Merge

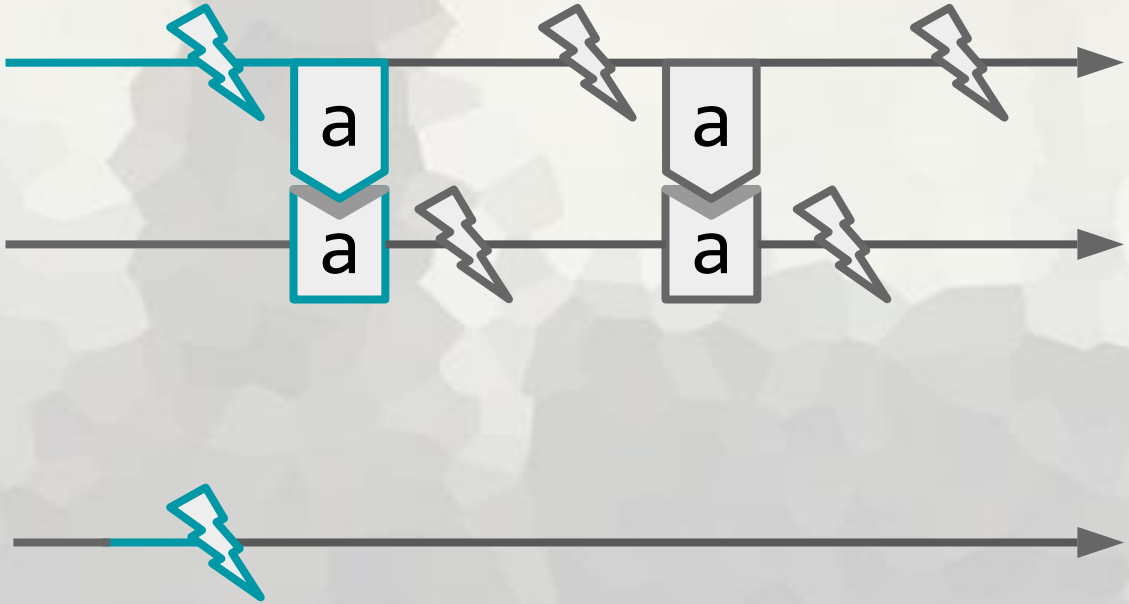
```
merge
```

Merge

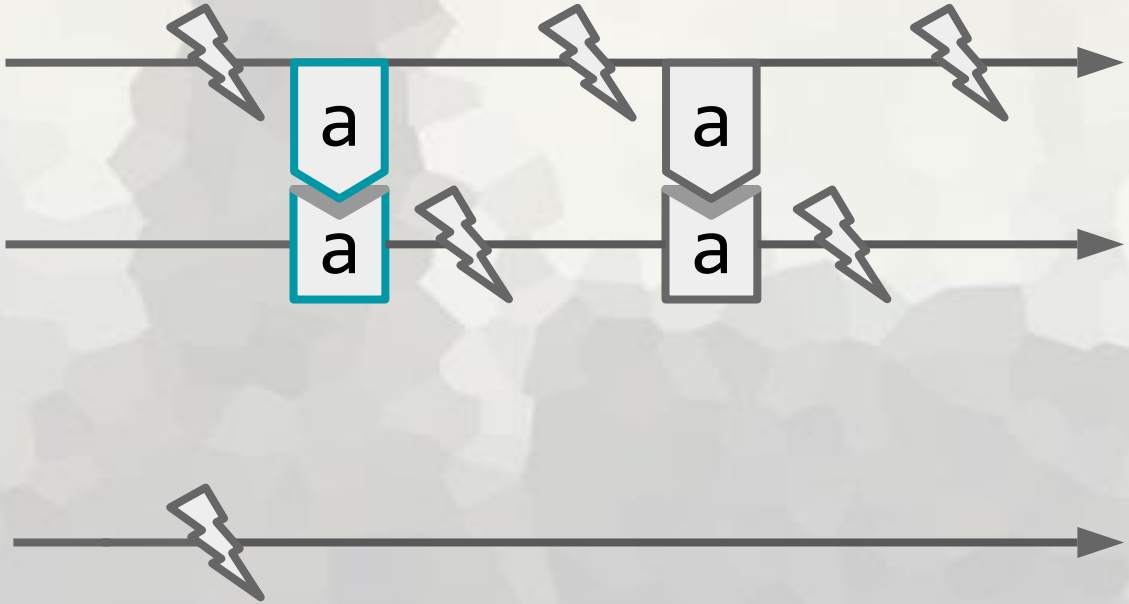
Merge



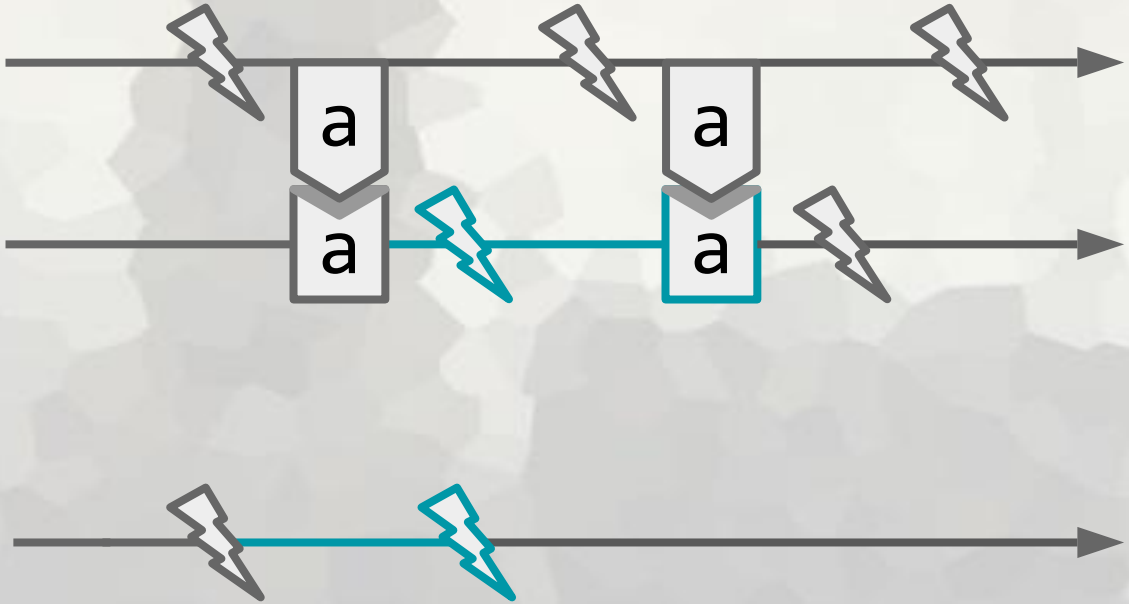
Merge



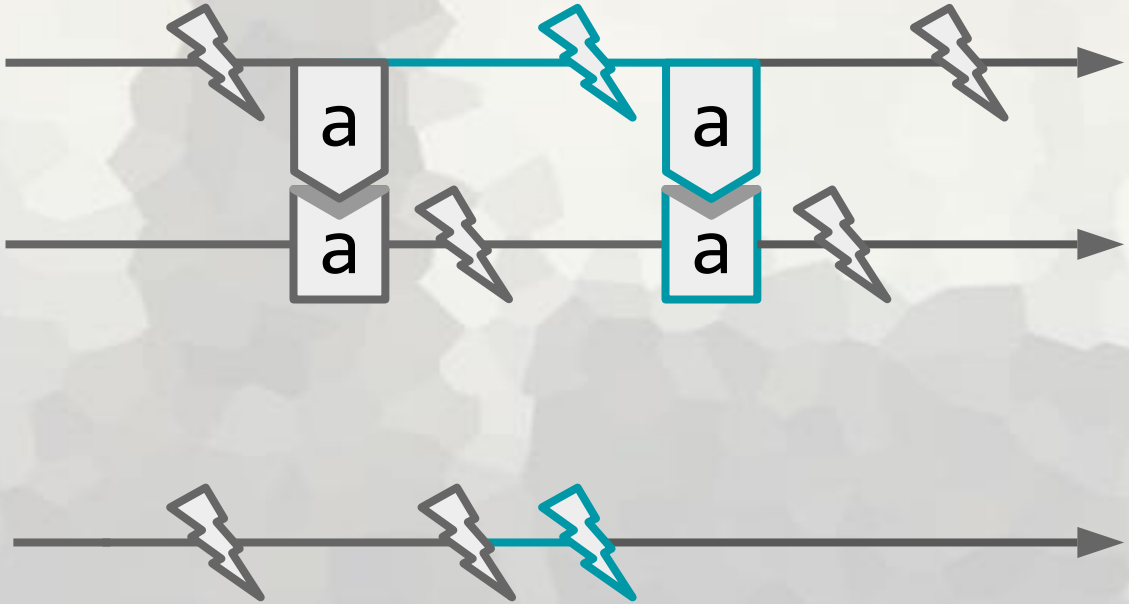
Merge



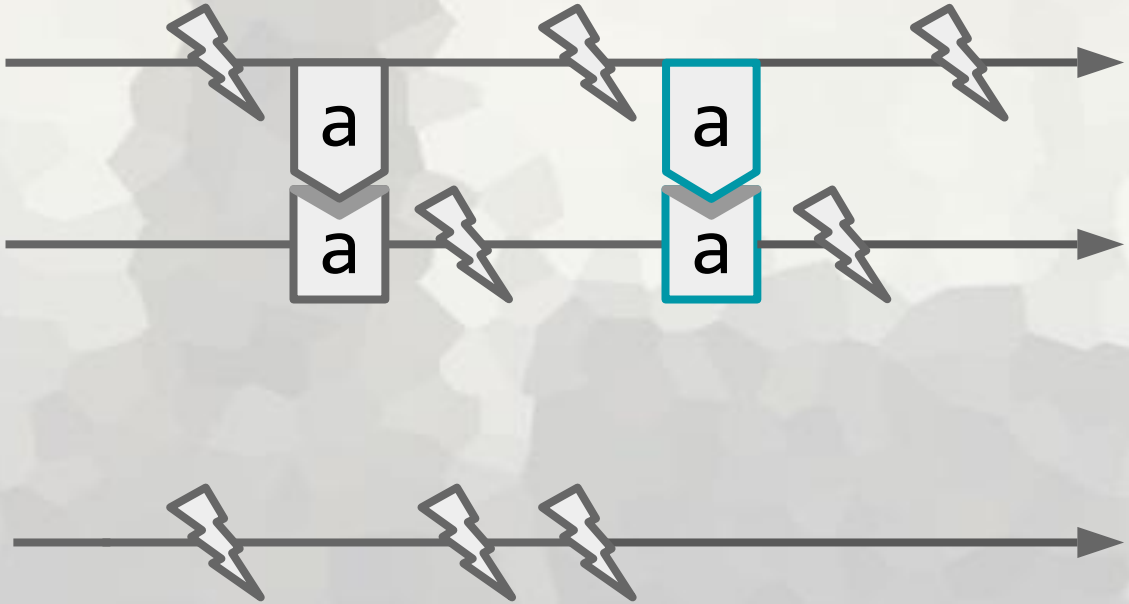
Merge



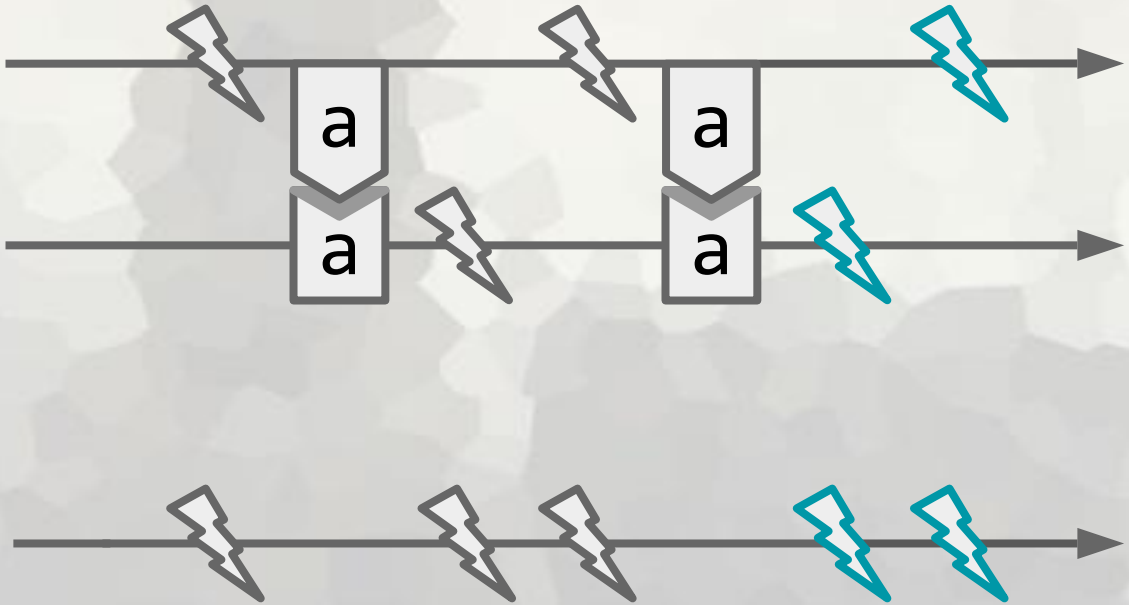
Merge



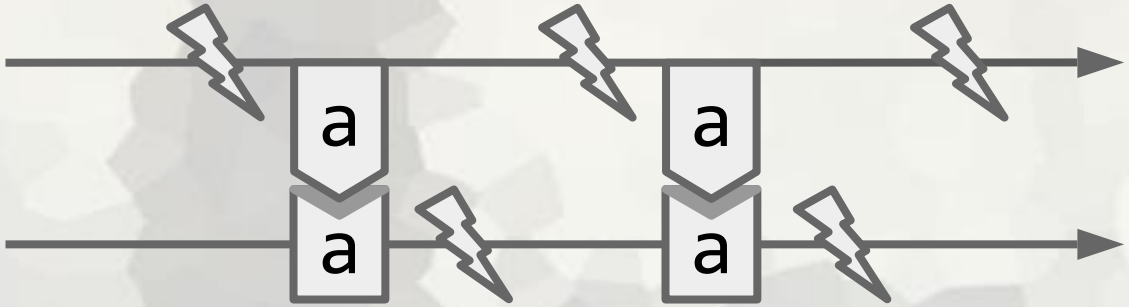
Merge



Merge



Merge



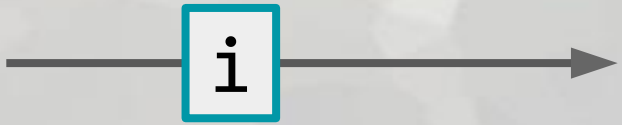
Result Step 1

Step 2: Filter Bad

Filter

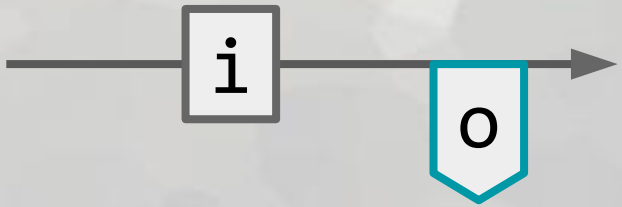
Pipe

P(ipe) i o m r



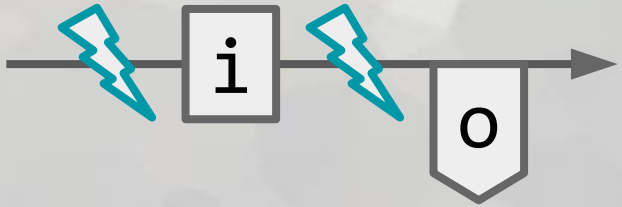
Pipe

P(ipe) i o m r



Pipe

P(ipe) i o m r



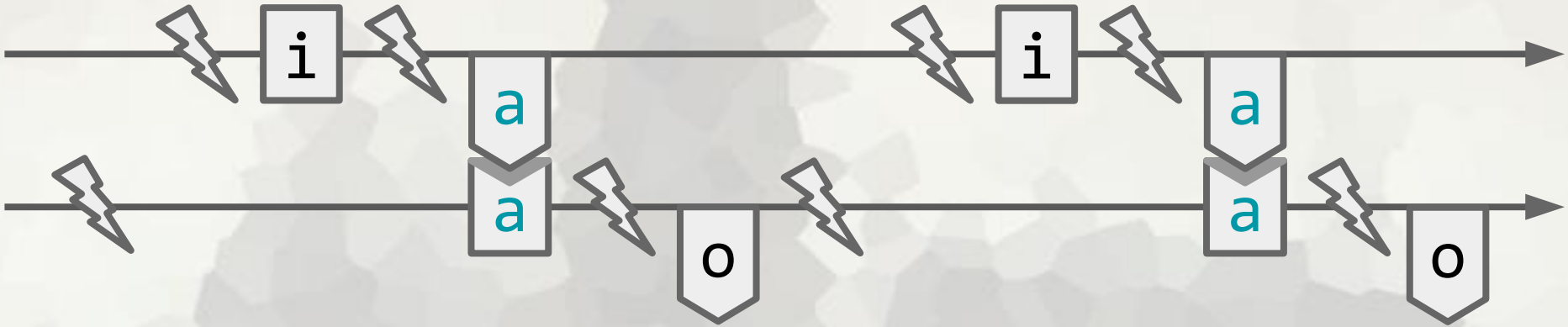
Pipe

```
Pr(oducer) o m r
= P(ipe) ∅ o m r
```

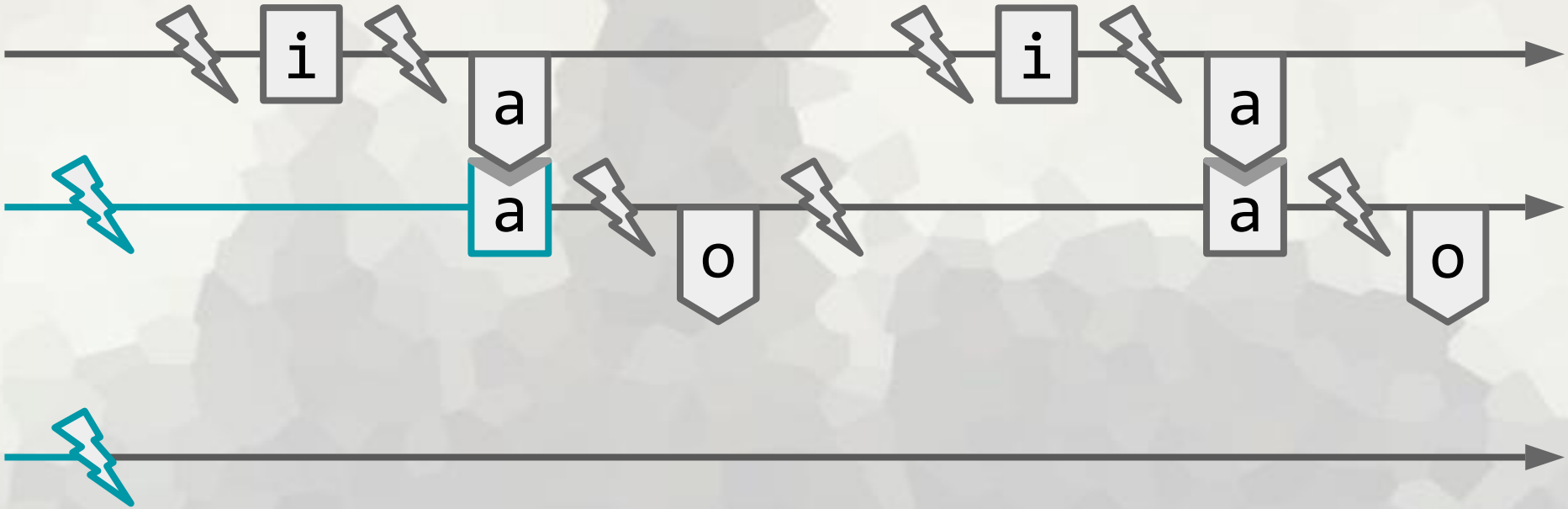
Pipe

Co(nsumer) o m r
= P(ipe) i ∅ m r

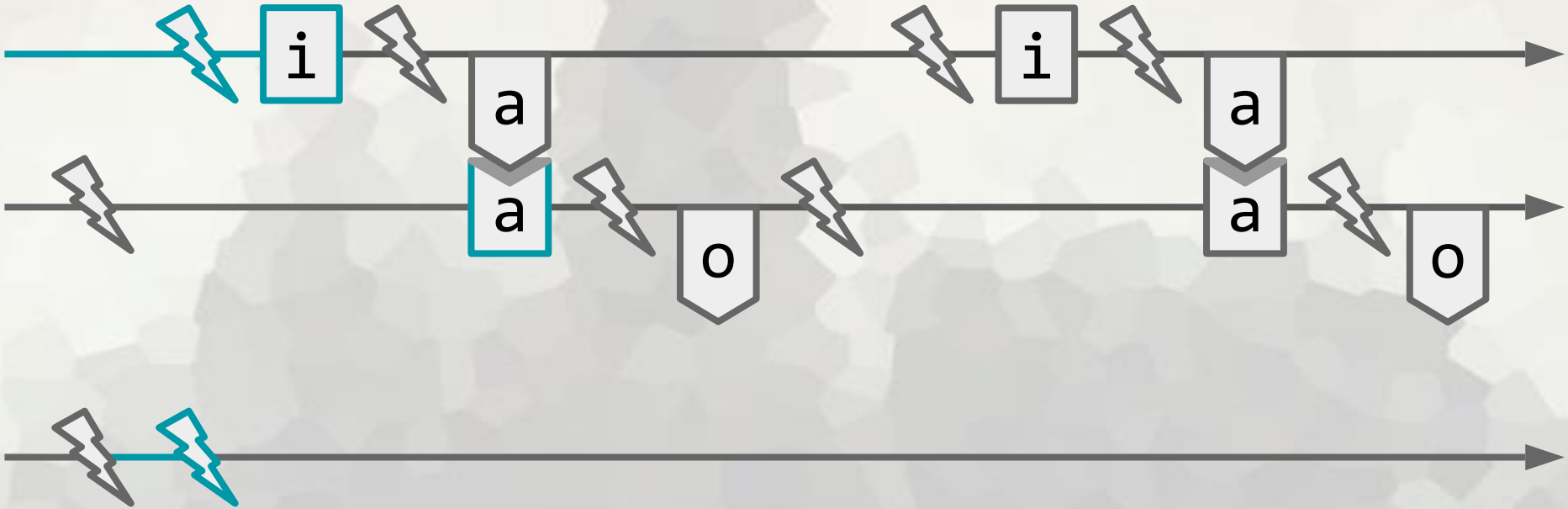
Merge



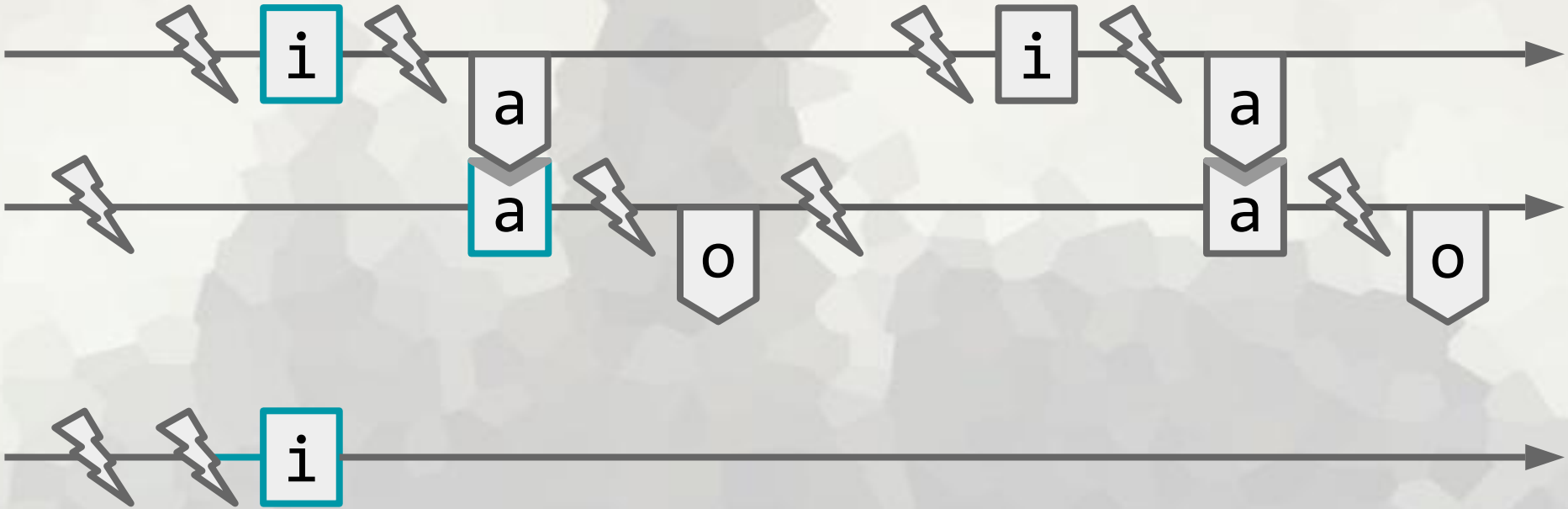
Merge



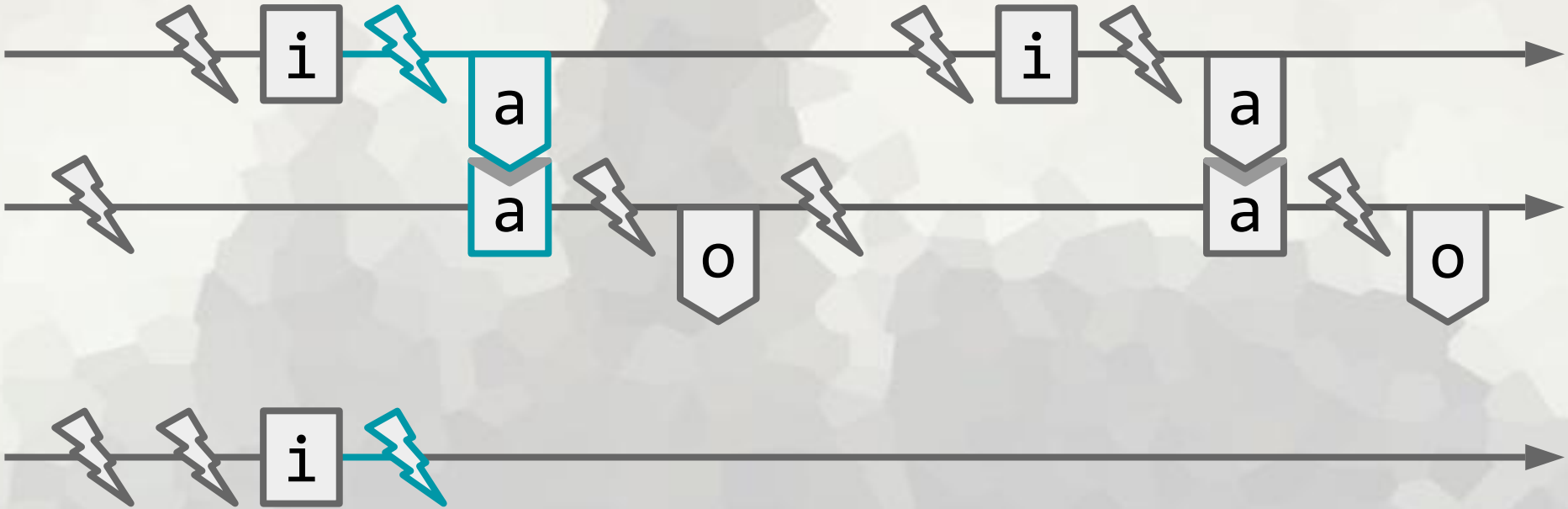
Merge



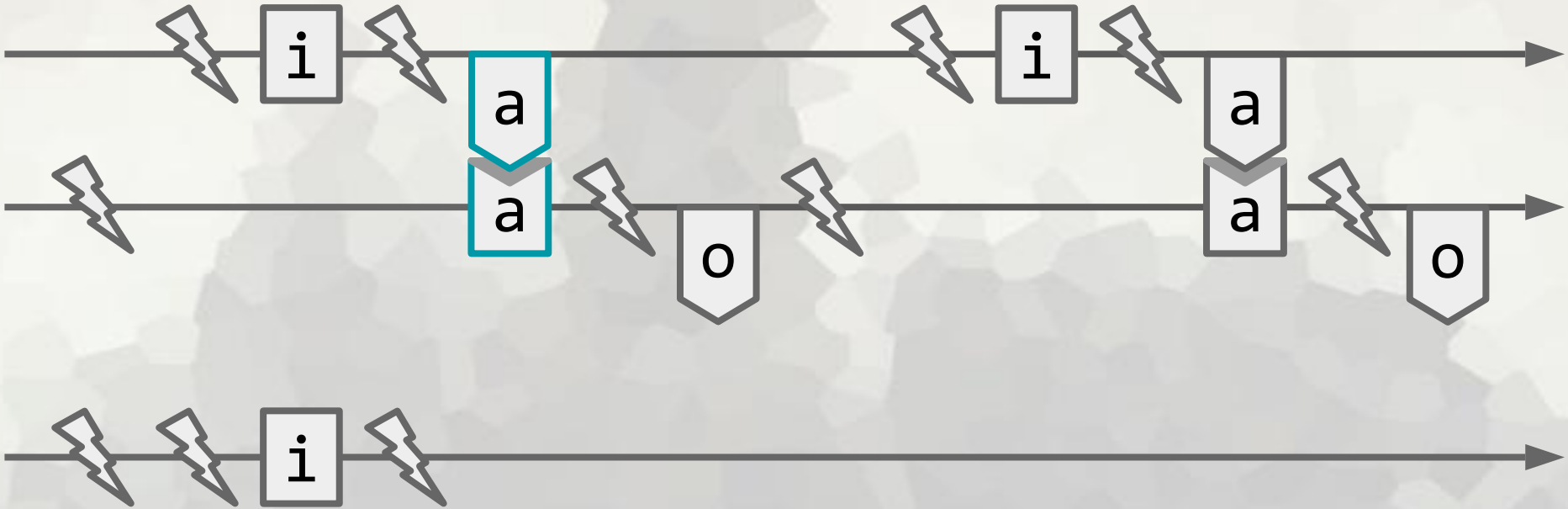
Merge



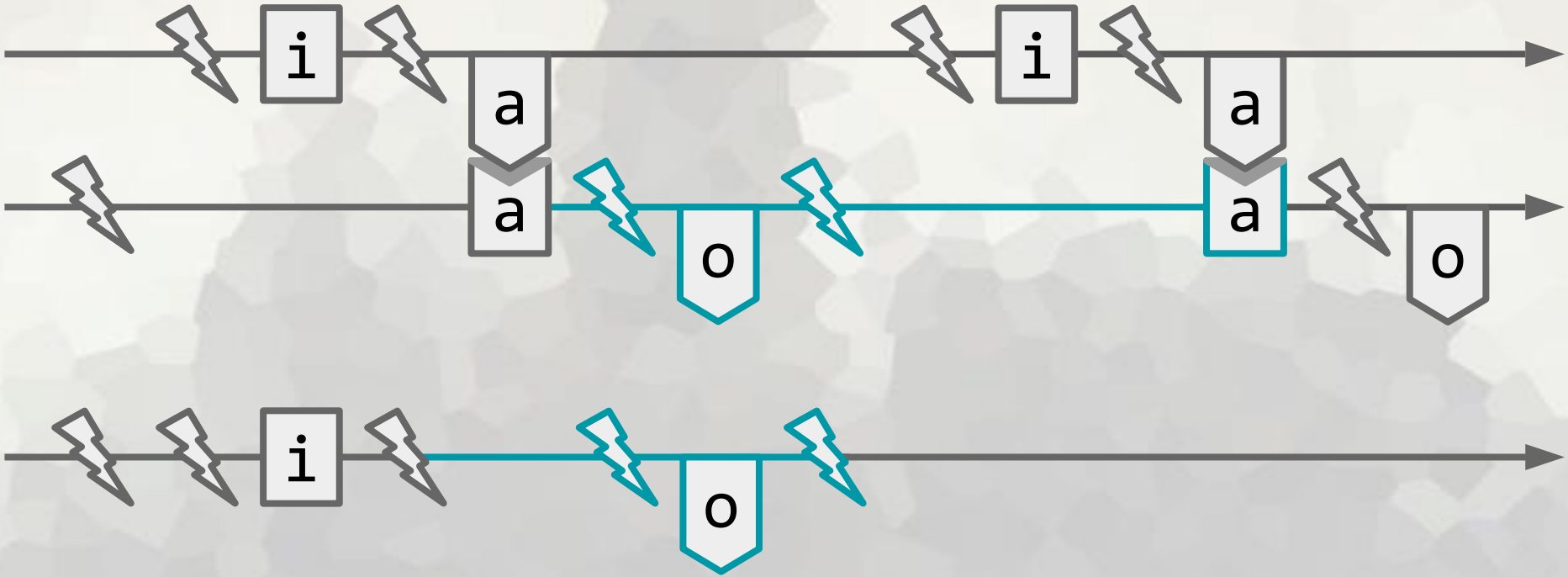
Merge



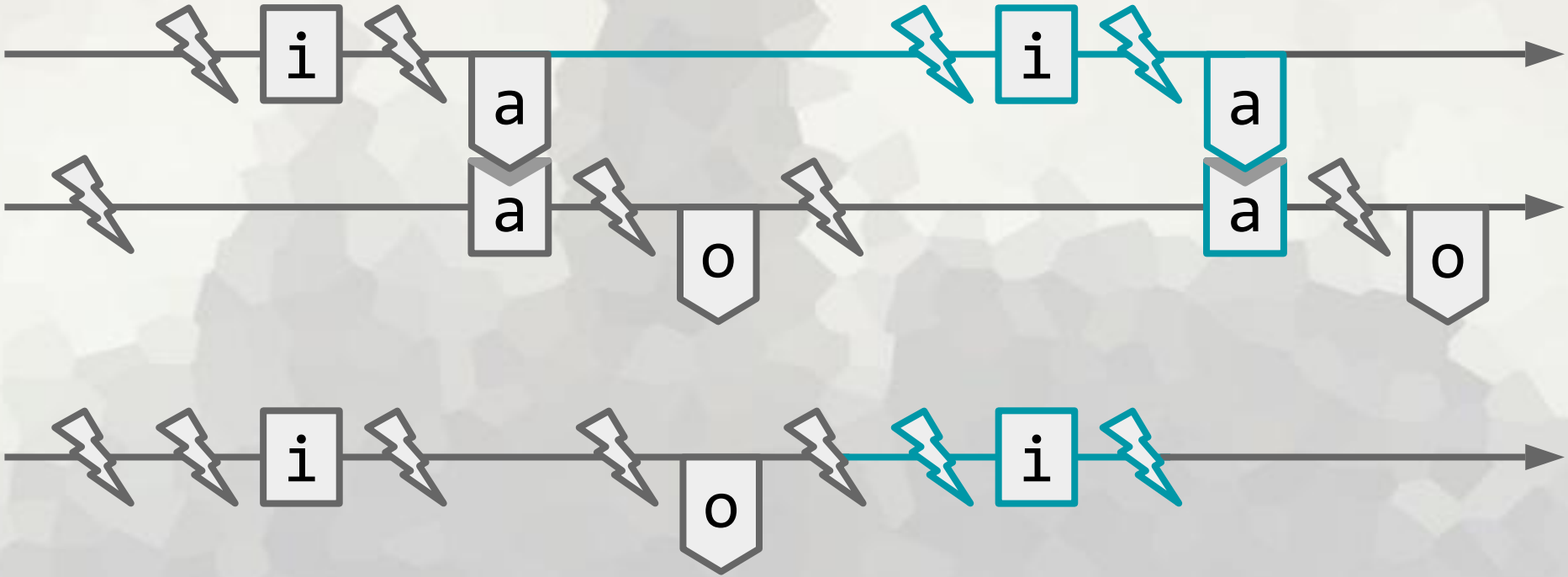
Merge



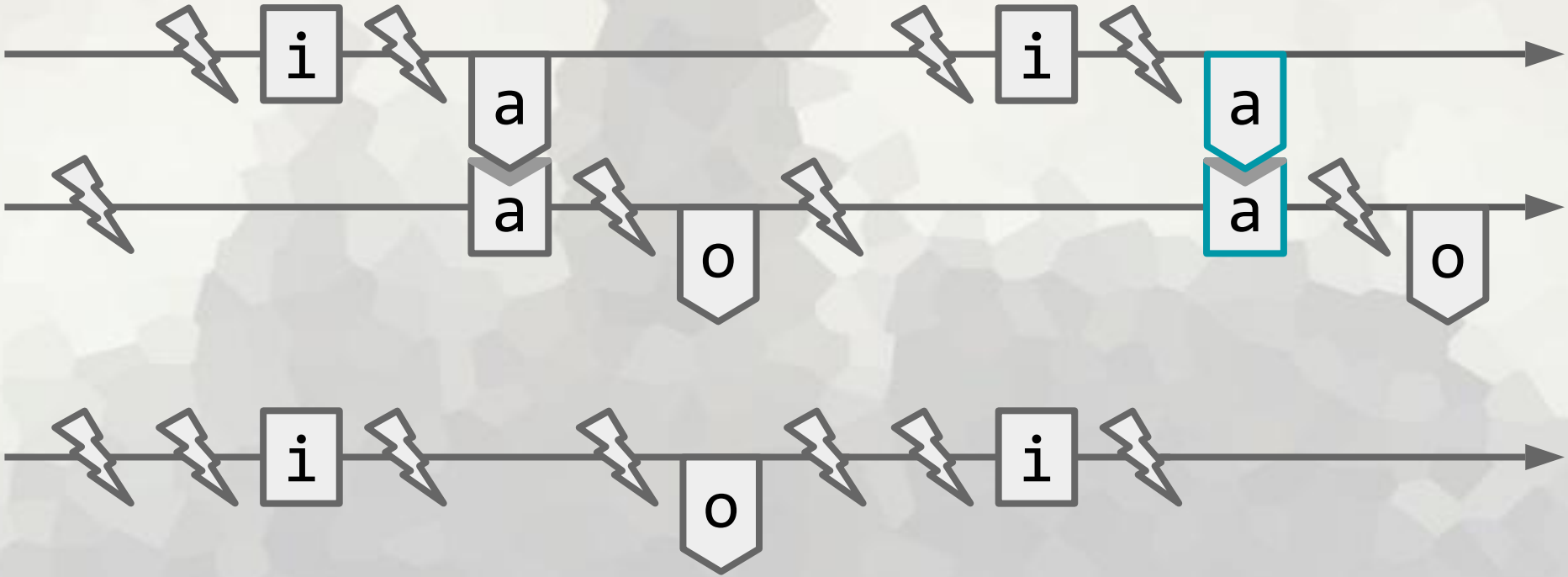
Merge



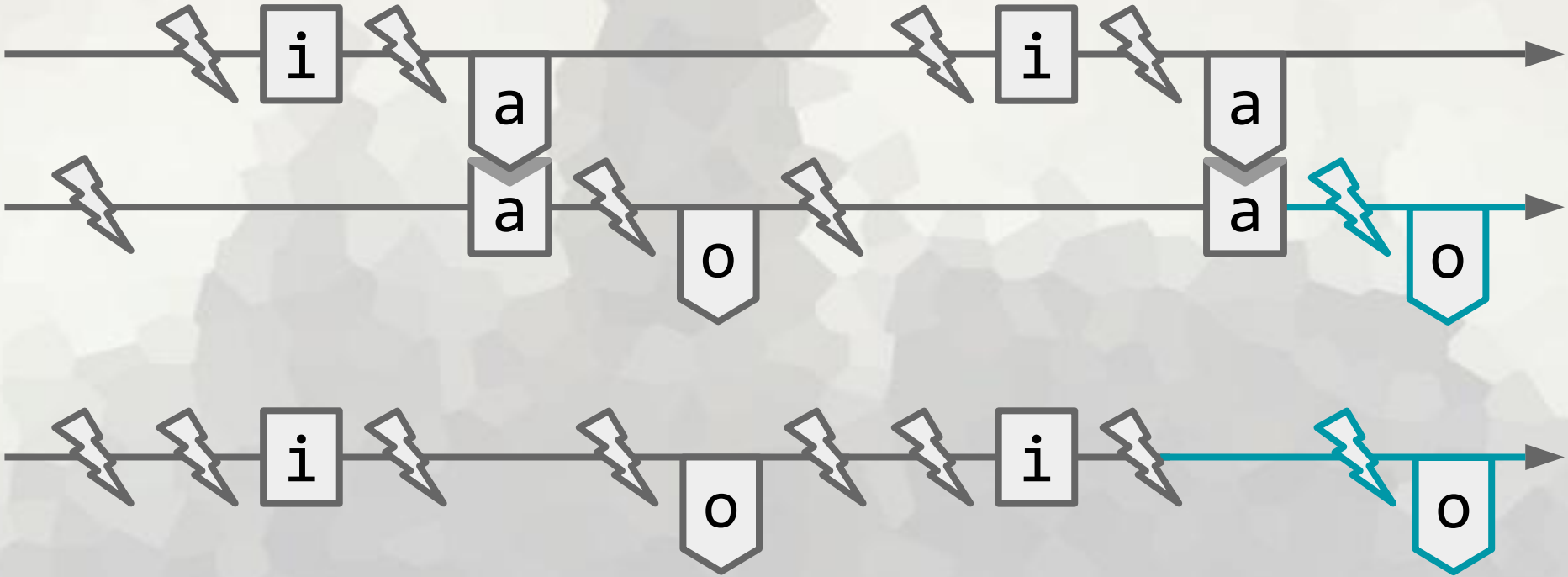
Merge



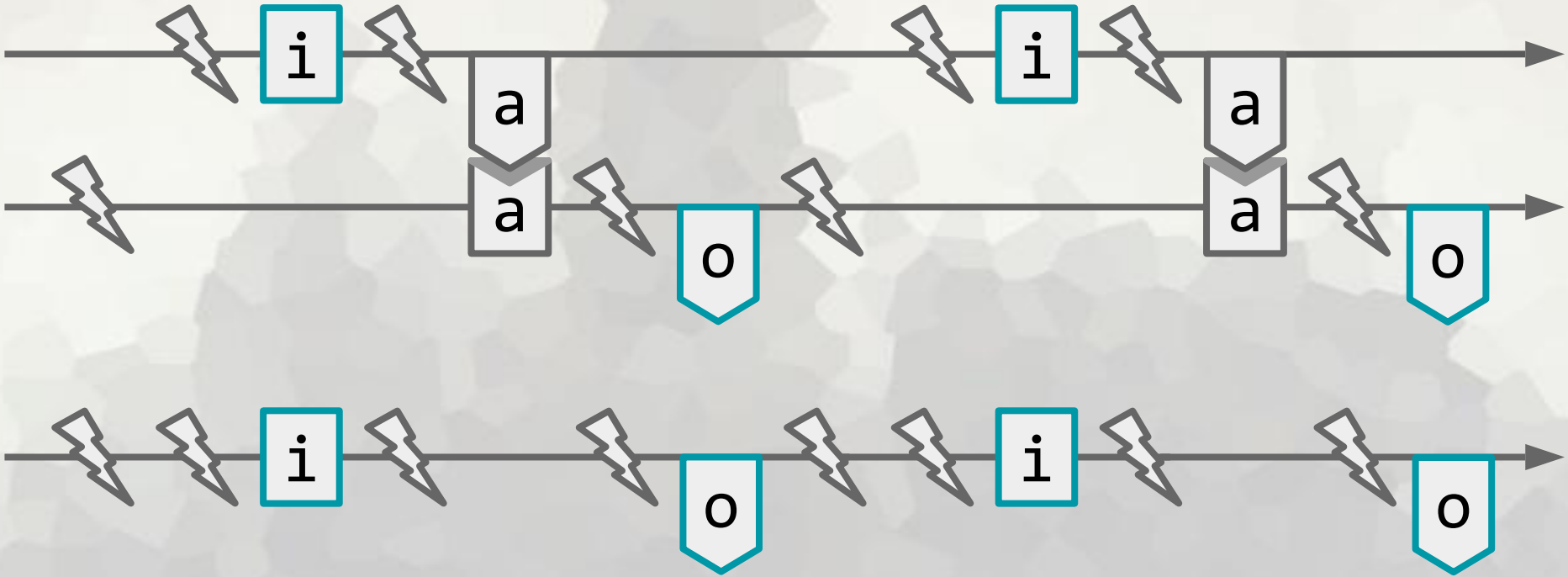
Merge



Merge



Merge



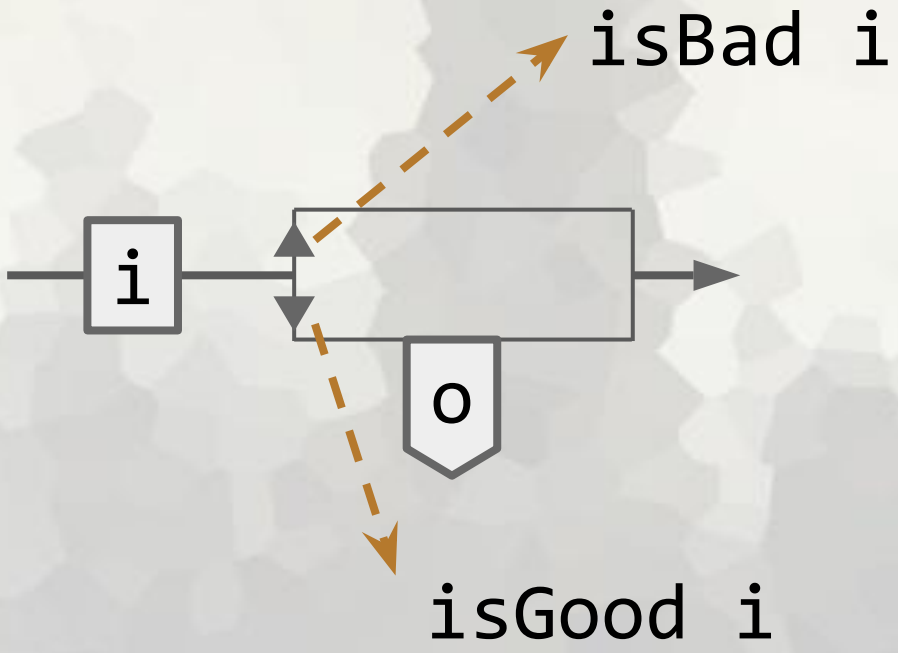
Merge

pipes: (>->)

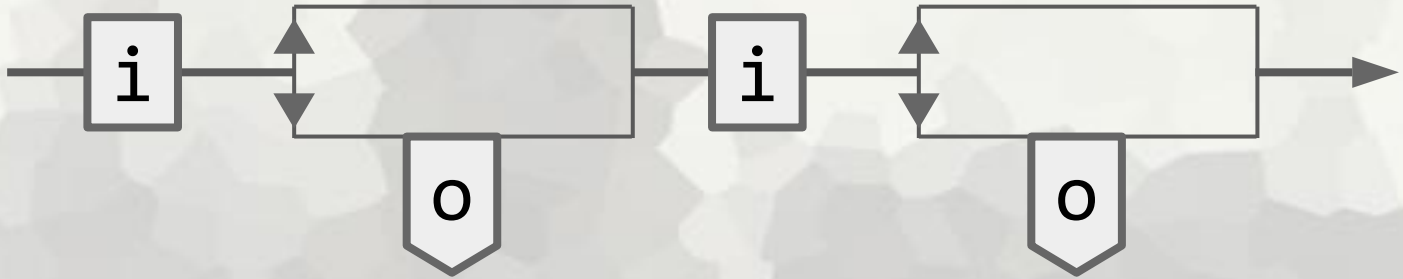
conduit: (.|)

Filter

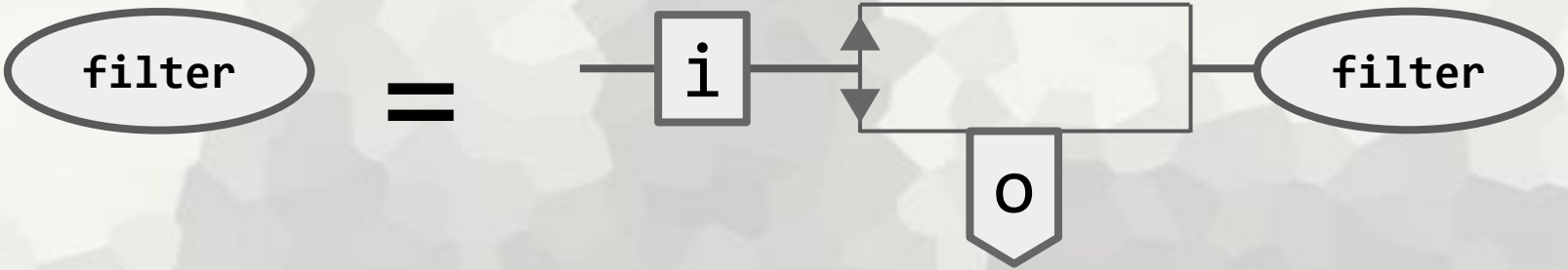
Filter



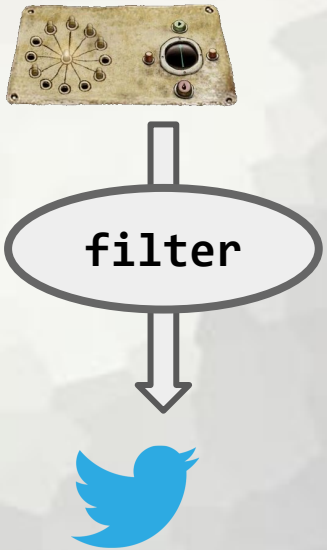
Filter

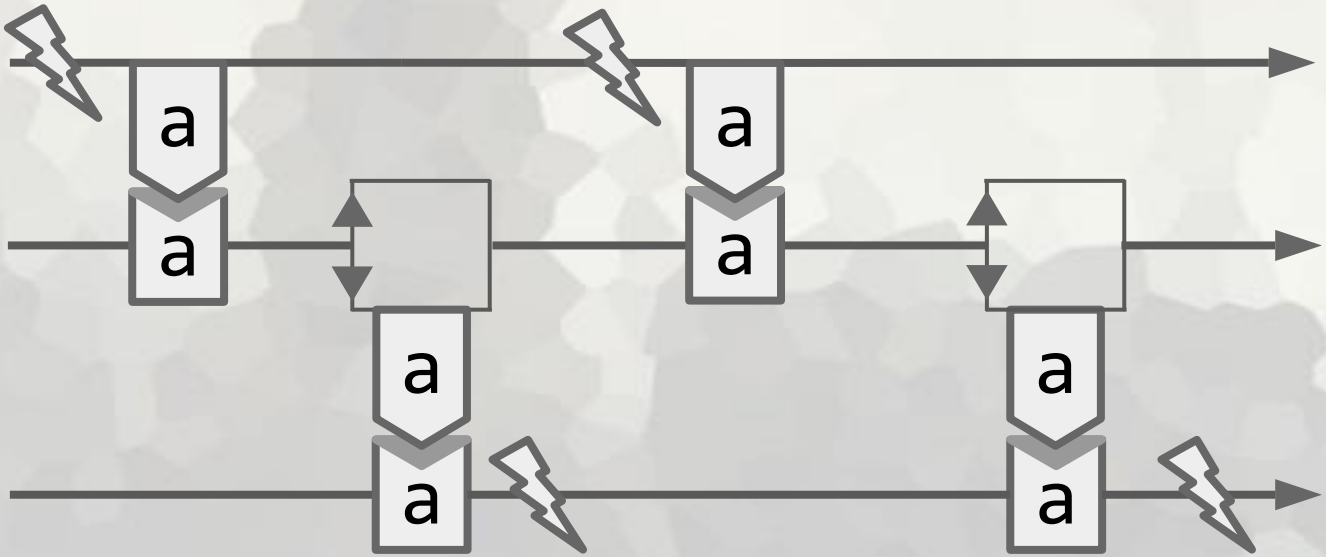


Filter

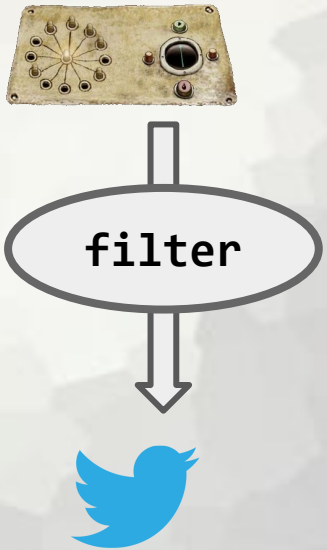


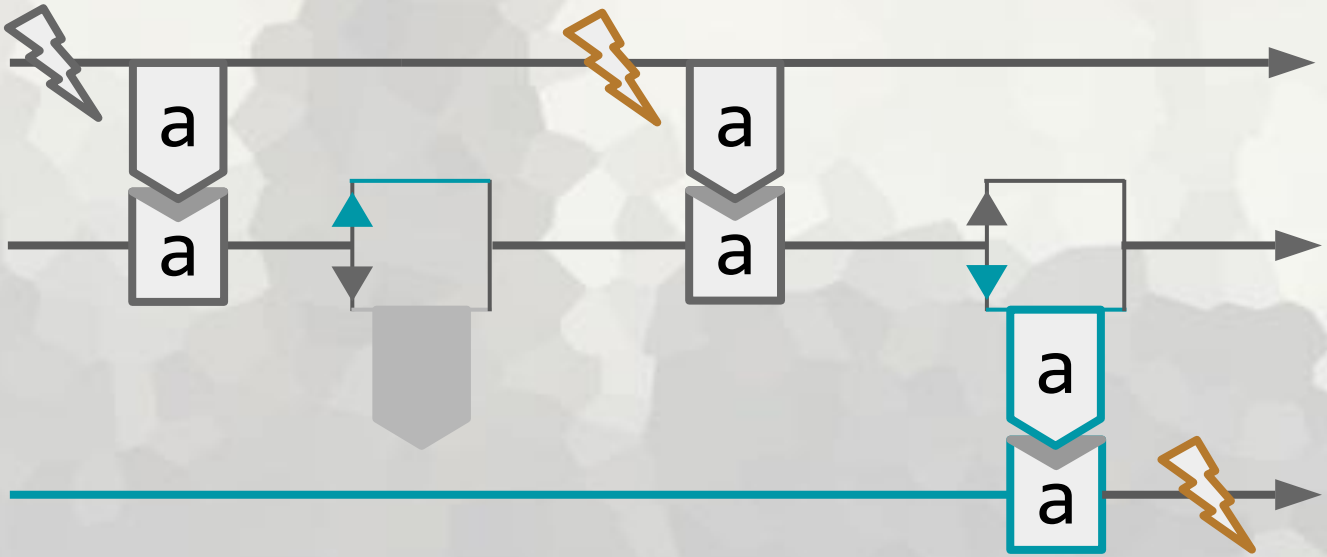
Filter



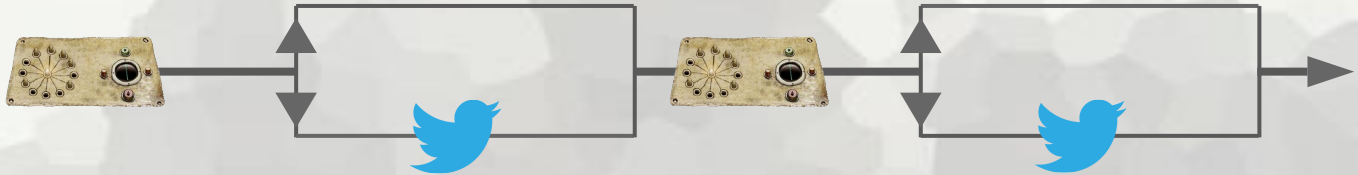


Filter

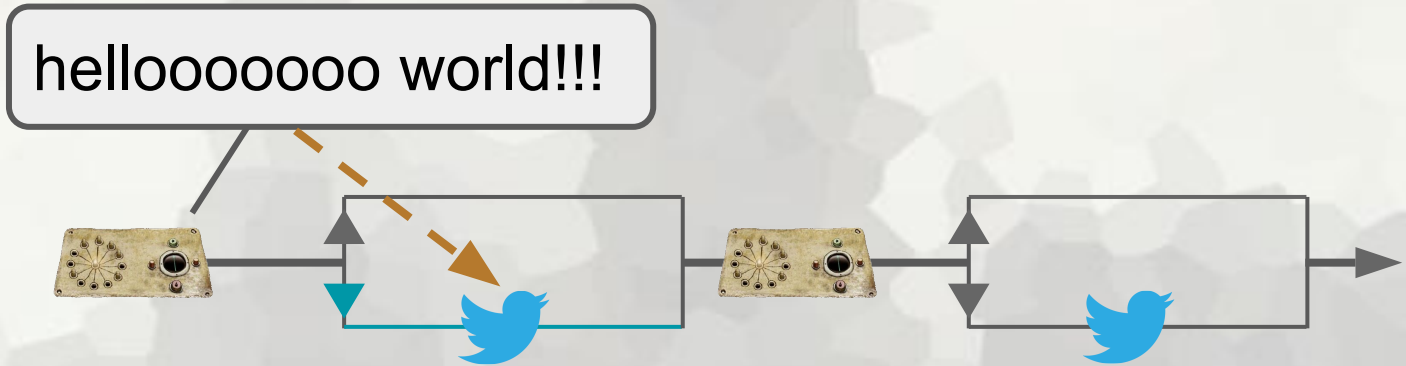




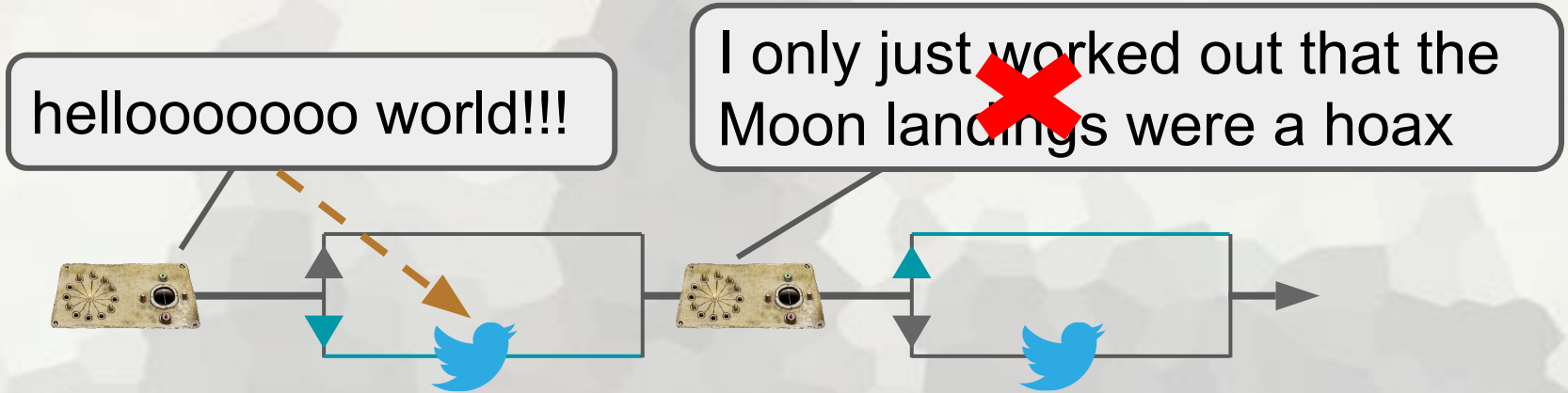
Result Step 2



Result Step 2



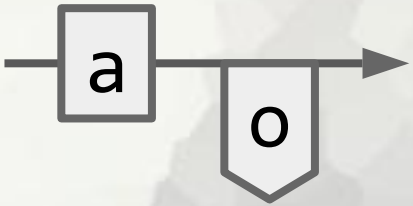
Result Step 2



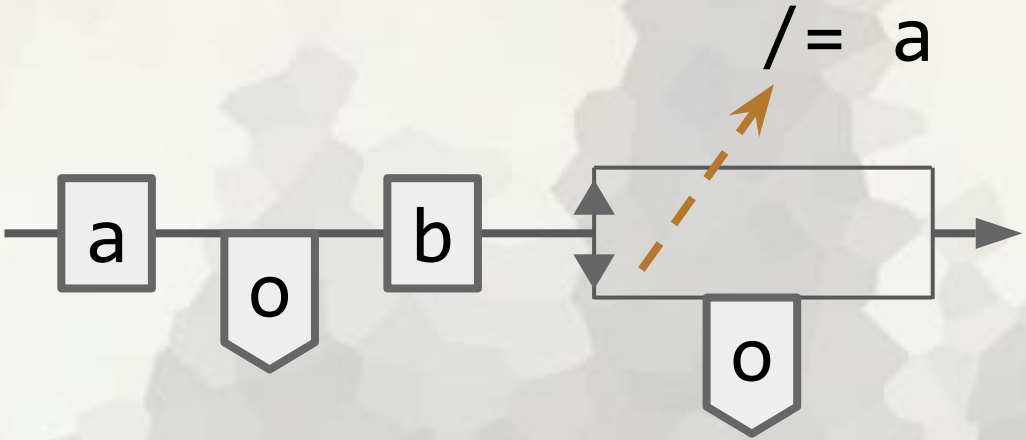
Step 3: Filter Duplicates

Nub

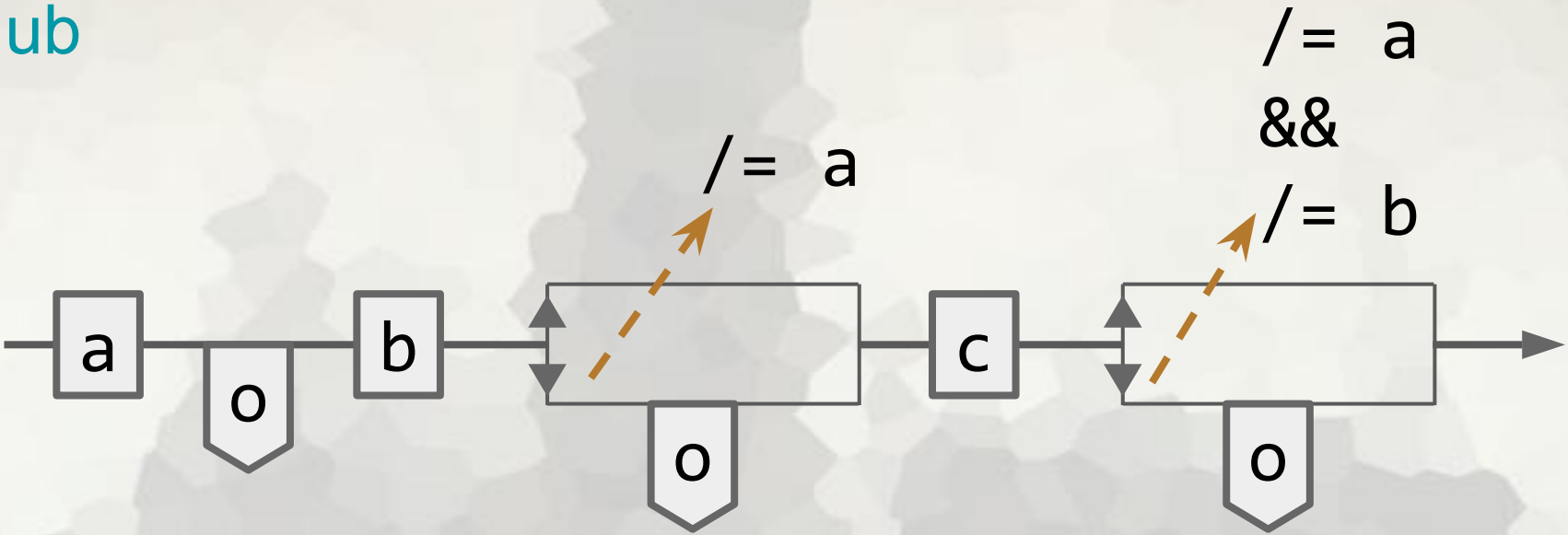
Nub



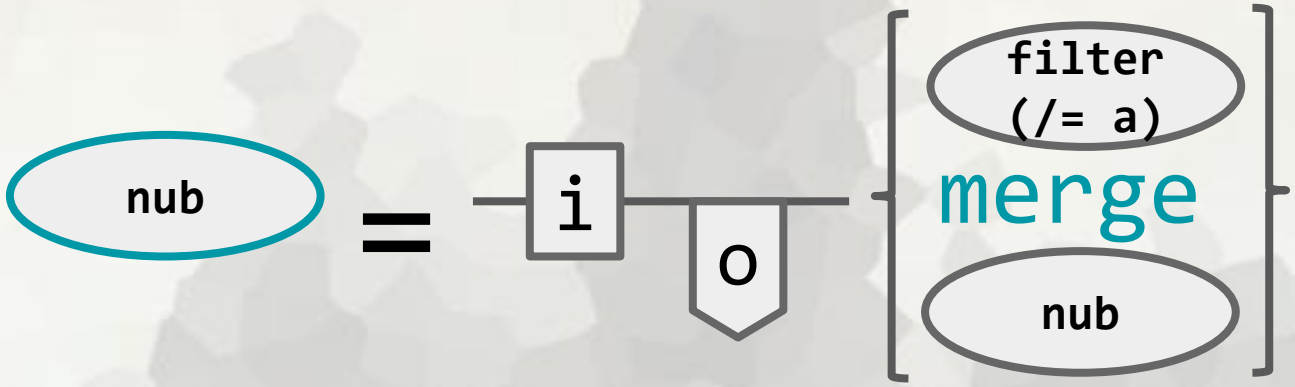
Nub



Nub



Nub



Result Step 3

Result Step 3

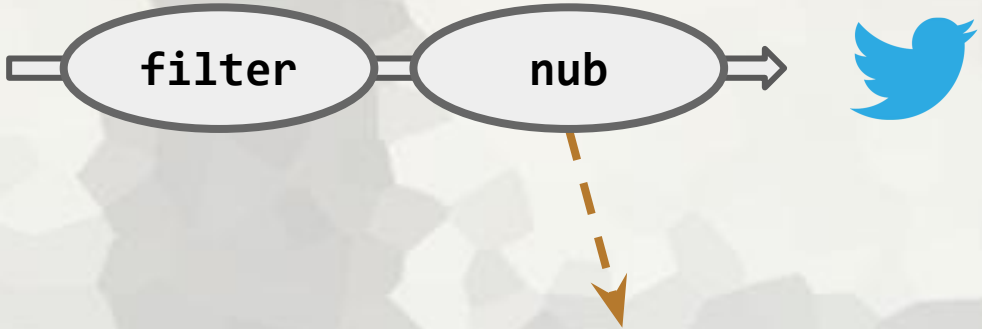
helloooooooo world!!!

im a nice person!

I only just worked out that the Moon landings were a hoax

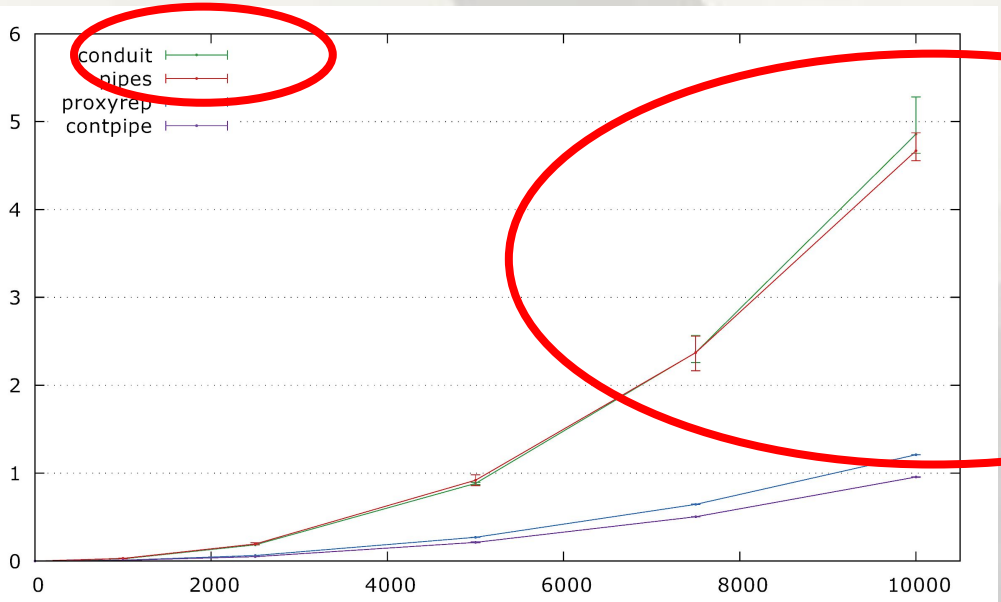
hellooooooo world!!!

Result Step 3



Contains many `merge` calls

Merge Performance

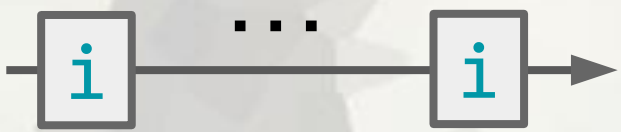


Alternative Representation

Traditional Representation


```
Pr o r = (o, … (o, Pr o r))
```

Traditional Representation




```
Co i r = (λi. … (λi. Co i r))
```

Alternative Representation


```
Pr o r: K_o -> r =
```

Alternative Representation

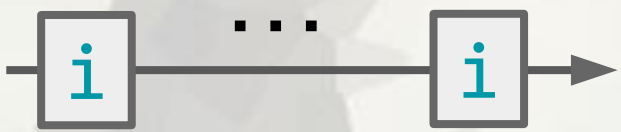

```
Pr o r: K_o -> r =
λk. k o ( ... (λk. k o (
   Pr o r)))
```

Callback

Alternative Representation


```
Pr o r: K_o -> r =
λk. k o ( ... (λk. k o (
   Pr o r)))
```

Alternative Representation

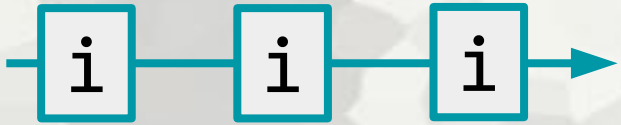



```
Co i r: K_i -> r =
λk. k (λi k. … (λi. Co i r))
```

Alternative Representation

```
merge = f x
where
```

f =



x =

Alternative Representation Pipe

?

Alternative Representation Pipe


```
P i o r: K_o -> K_i -> r =
```

Alternative Representation Pipe


```
P i o r: K_o -> K_i -> r =
λk_o k_i. k_i (λi k_o k_i. k_o o
   (P i o r))
```

Alternative Pipe Merge

```
mergeP = f x
where
   f :  K_a -> K_o -> r

   x :  K_i -> K_a -> r
```

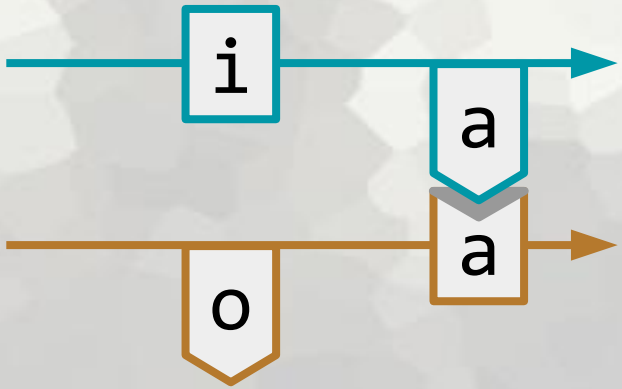
Alternative Pipe Merge

```
mergeP      f x
where
   f :                    -> r

   x :   K_i -> K_a -> r
```

Alternative Pipe Merge

```
mergeP = ? p q
where
  p =
  q =
```



Alternative Pipe Merge

```
mergeP = _ : K_i -> K_o -> r
where
  p : K_i -> K_a -> r

  q : K_a -> K_o -> r
```

Alternative Pipe Merge

```
mergeP = _ : K_i -> K_o -> r
where
  p : K_i -> K_a -> r

  q : K_a -> K_o -> r
```

Alternative Pipe Merge

```
mergeP = λk_o k_i. _ : r
where
  p : K_i -> K_a -> r

  q : K_a -> K_o -> r
```

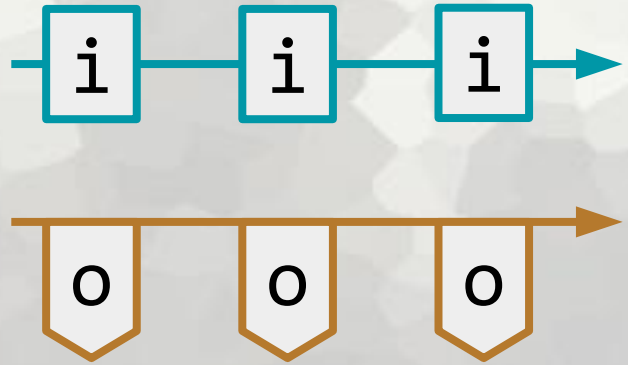
Alternative Pipe Merge

```
mergeP = λk_o k_i. _ : r
where
  λx. p k_i x :        K_a -> r

  λx. p x k_o : K_a ->        r
```

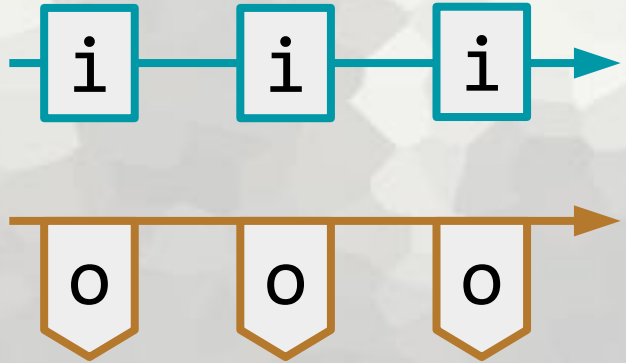
Alternative Pipe Merge

```
mergeP = merge p' q'
where
```

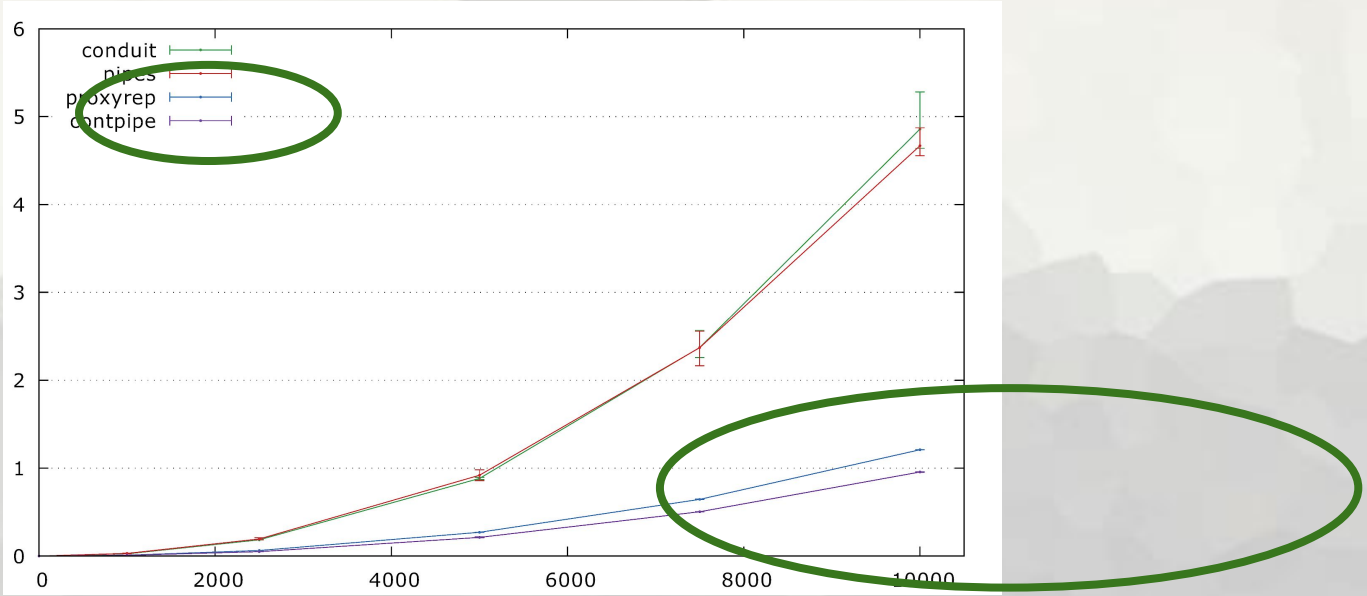


Alternative Pipe Merge

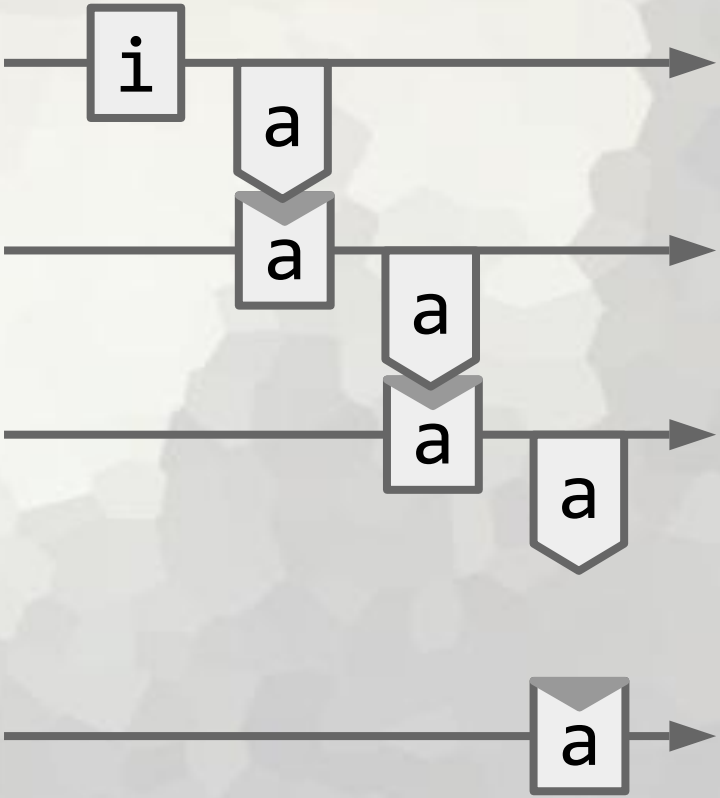
```
mergeP = p' q'
where
```



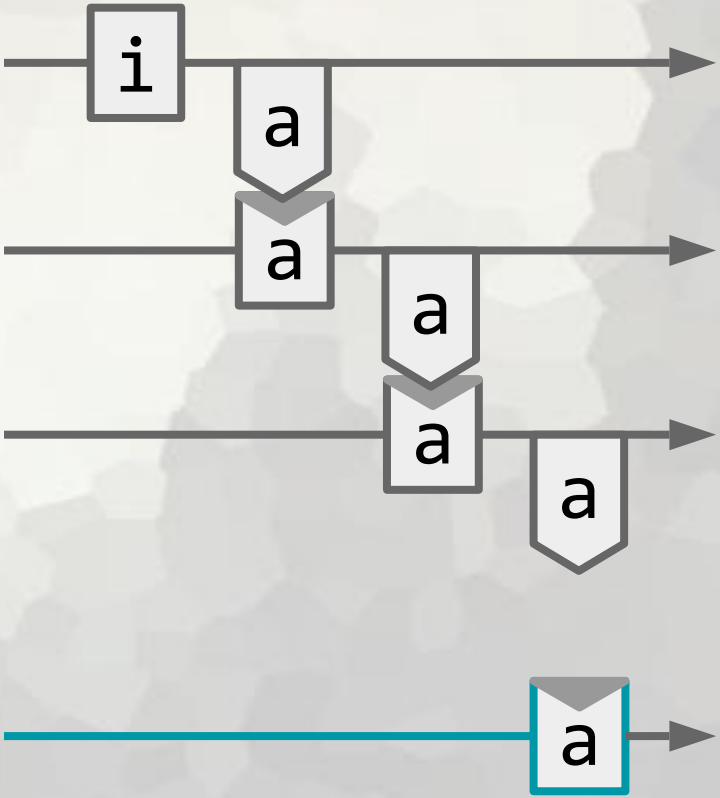
Alternative Pipe Merge



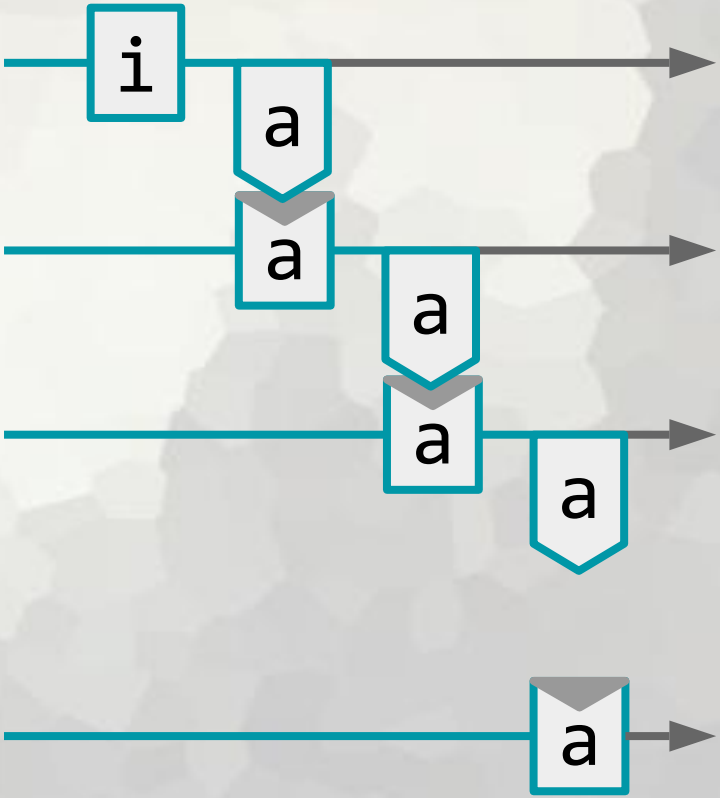
Nested Traditional Merge



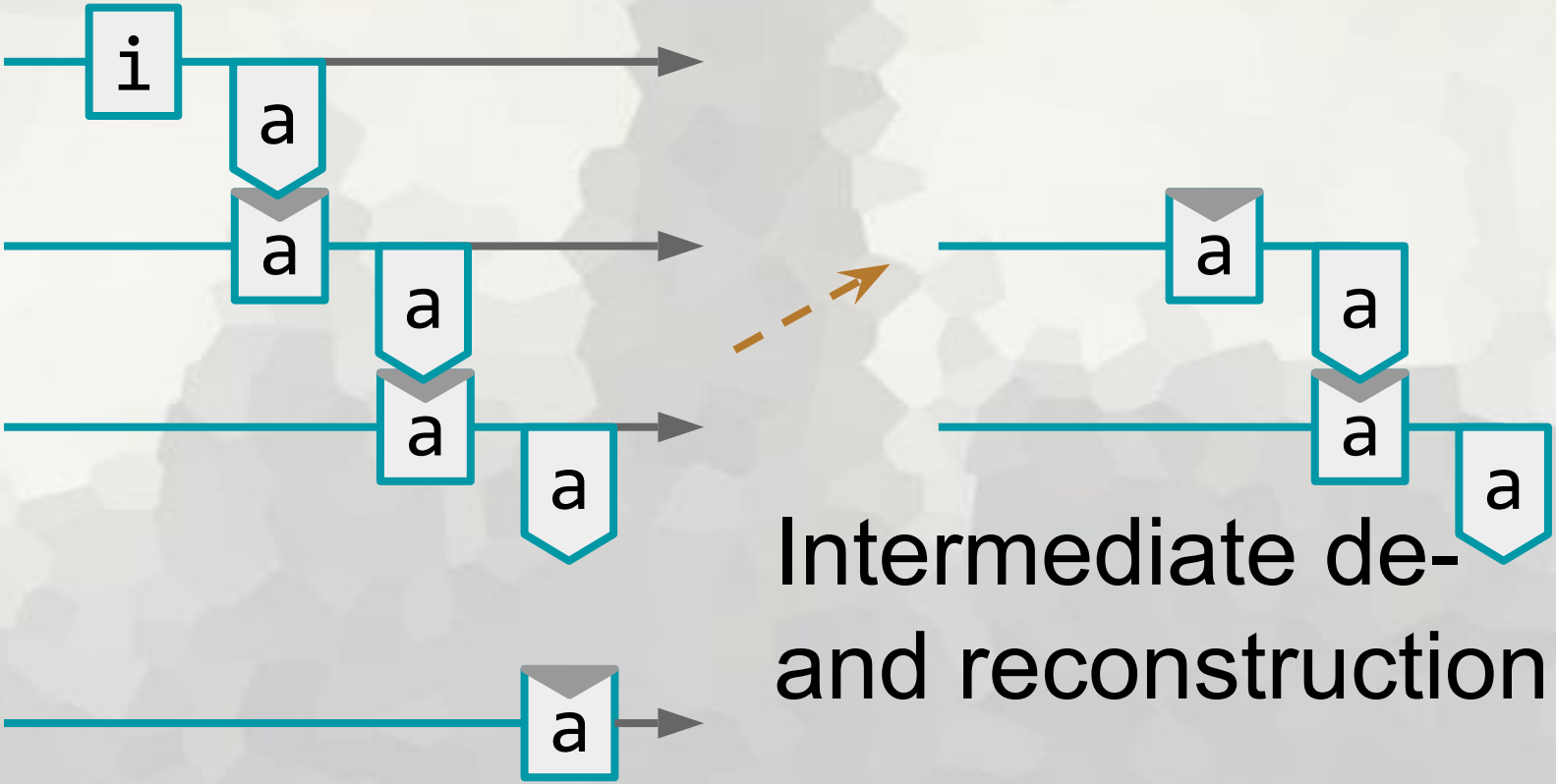
Nested Traditional Merge



Nested Traditional Merge

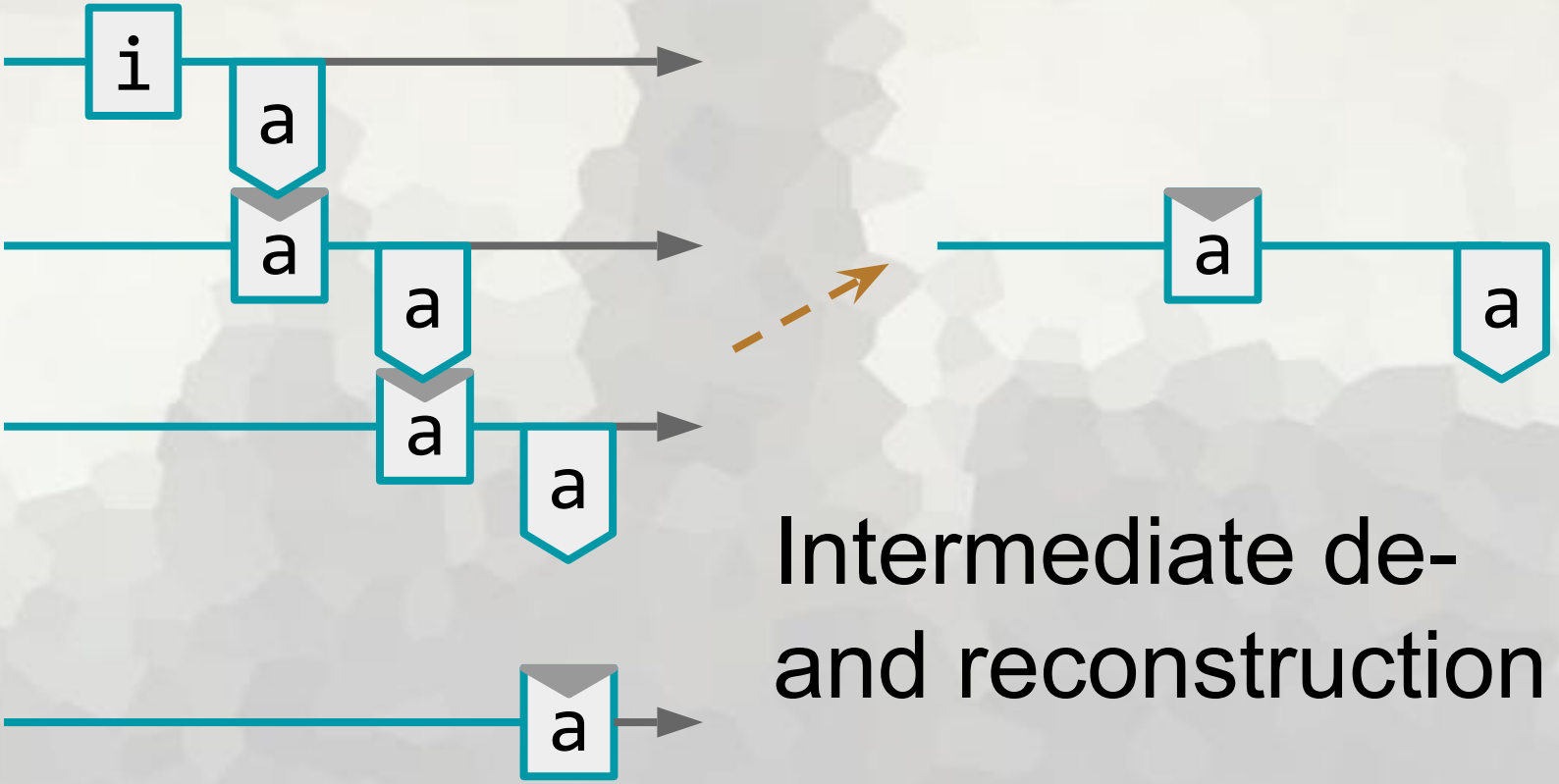


Nested Traditional Merge



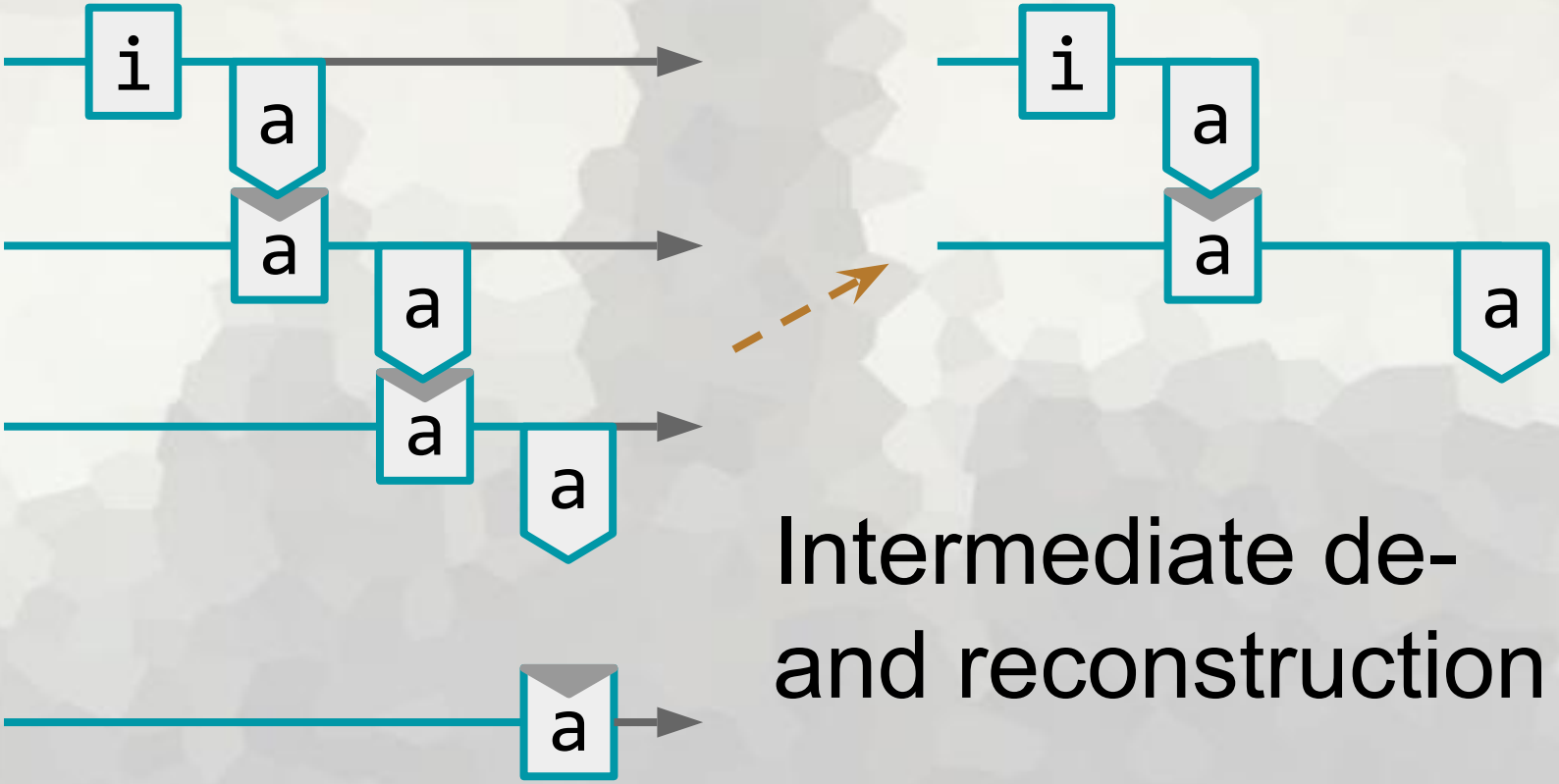
Intermediate de- and reconstruction

Nested Traditional Merge



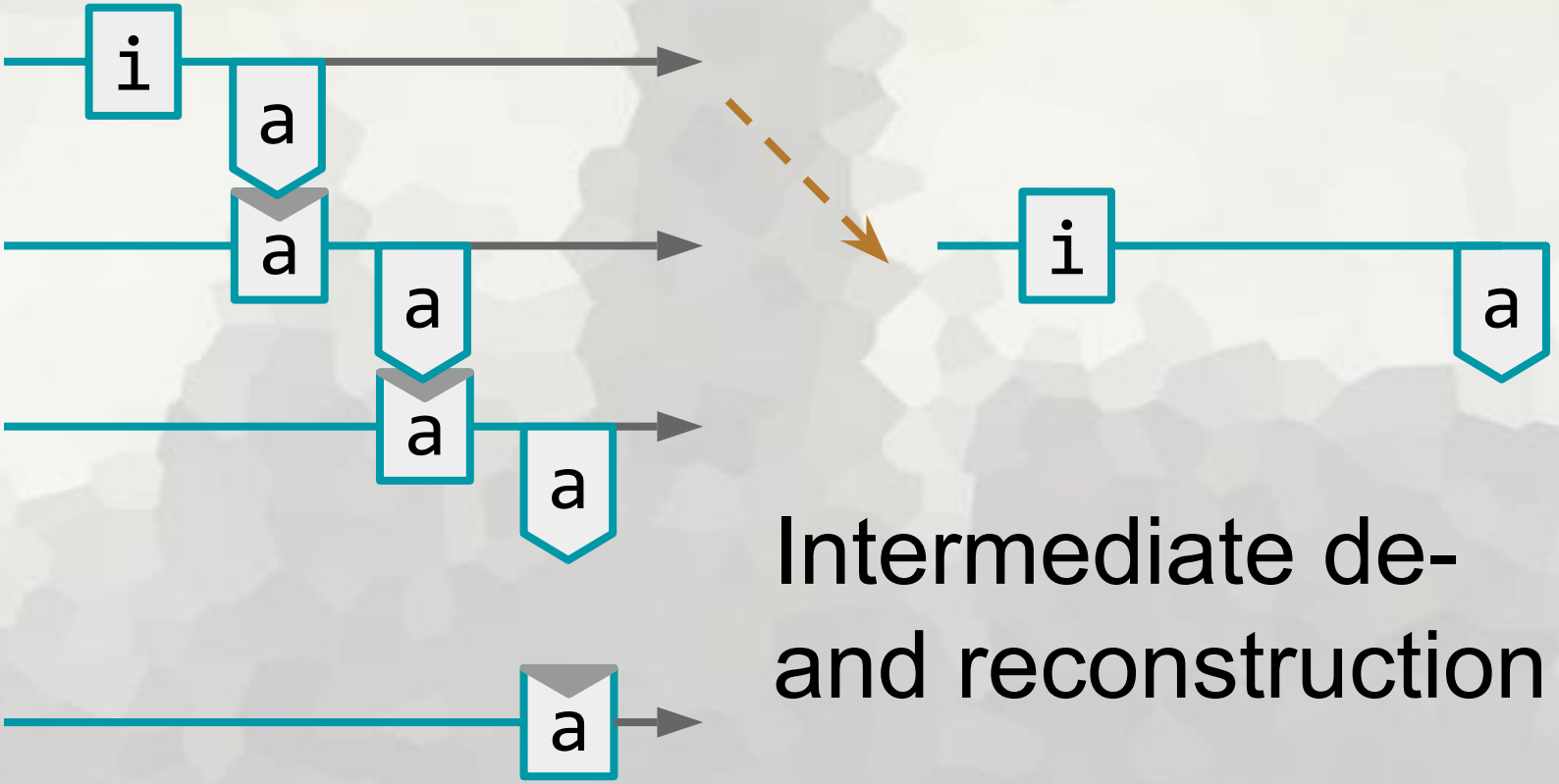
Intermediate de- and reconstruction

Nested Traditional Merge



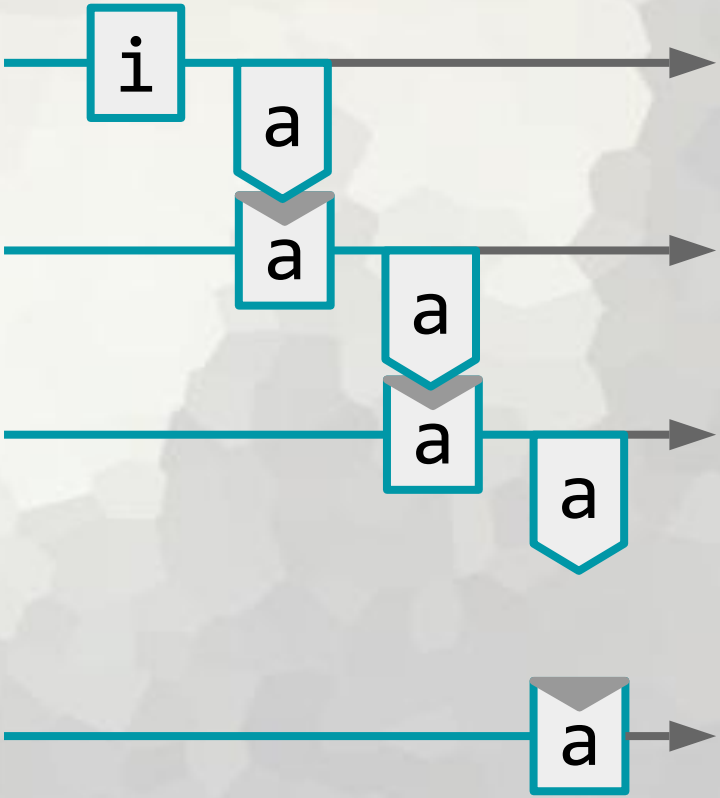
Intermediate de- and reconstruction

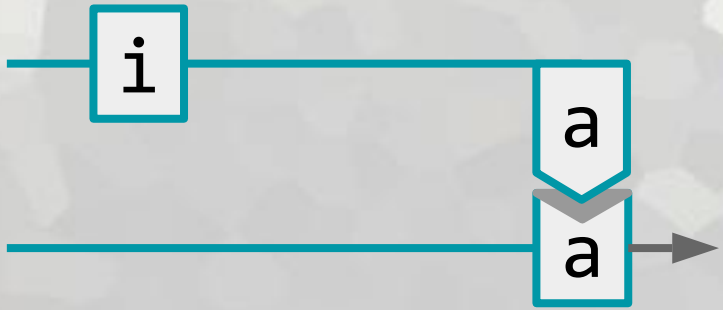
Nested Traditional Merge



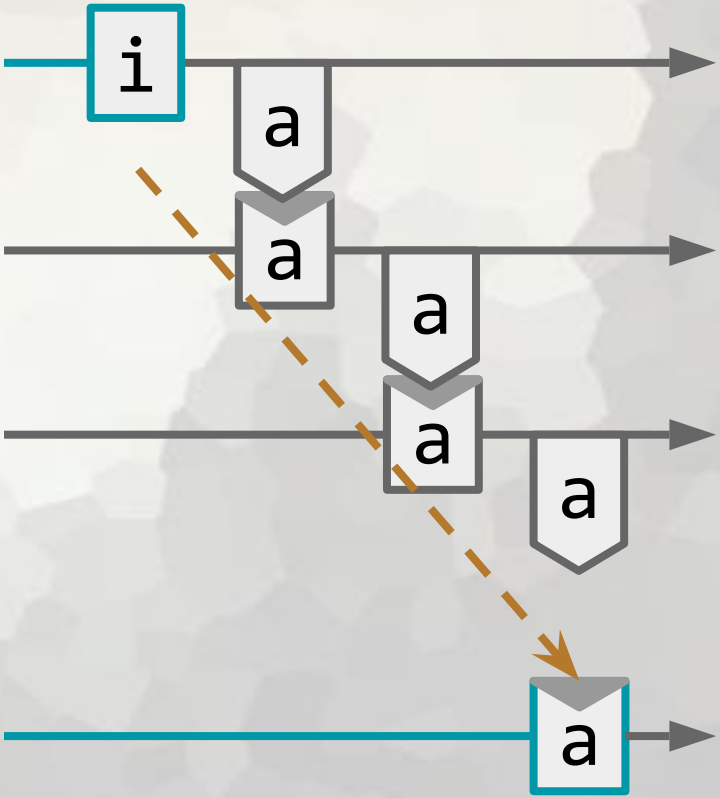
Intermediate de- and reconstruction

Nested Traditional Merge



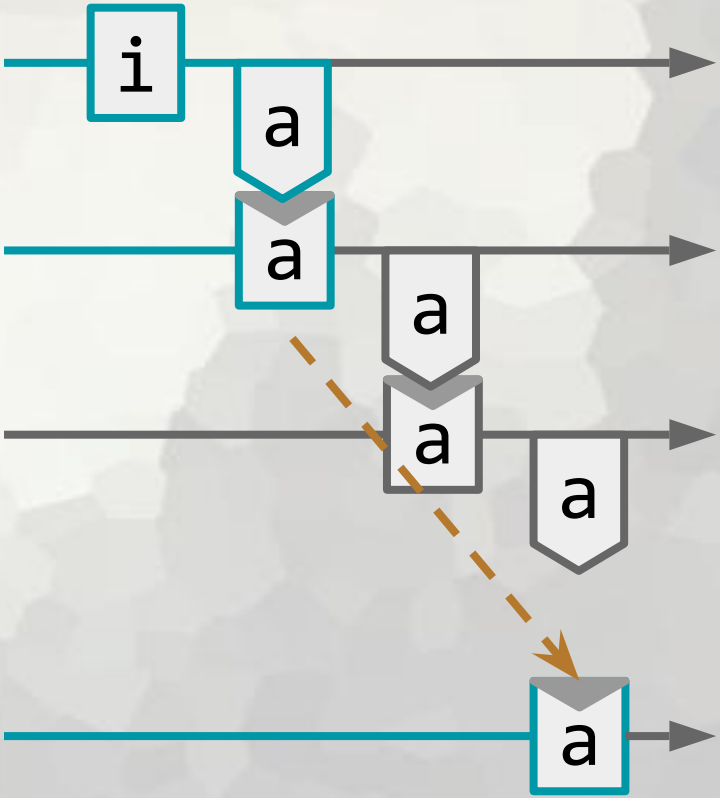


Nested Alternative Merge



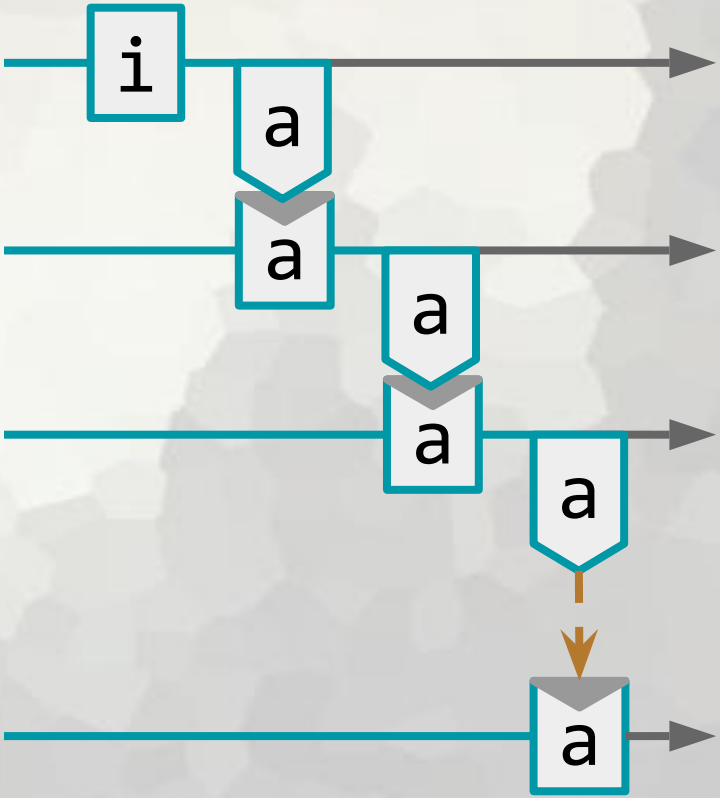
Pass the callback through the nested merge

Nested Alternative Merge



Pass the callback through the nested merge

Nested Alternative Merge



Pass the callback through the nested merge

Our Paper

How can we derive the fast representation from the traditional one systematically?

ruben.pieters@cs.kuleuven.be

some images from *Machinarium*