

The background of the slide is a photograph of a modern architectural site at TU Delft. In the upper center, a tall, grey, conical structure with a metal lattice framework rises into a clear blue sky. Below this, a large, wide set of concrete steps leads down a grassy slope. Numerous people are seen sitting and walking on these steps. To the right of the steps, a paved path with rectangular tiles leads across a green lawn where more people are gathered. The overall scene is bright and sunny, with green trees visible on the left and right edges.

# A type system for dynamic instances

## Delft University of Technology

Albert ten Napel

July 30, 2019

# Outline

- ① What are effects
- ② Algebraic effects and handlers
- ③ Miro
- ④ Semantics
- ⑤ Type system  
Type safety

# Effects

## Example

$x = 10$

$5 - 6$

# Miro - Creating instances

## Example

```
ref : forall s. Int -> (Inst s State){s}
ref [s] v =
  new State@s {
    get () k -> \st -> k st st
    put st' k -> \st -> k () st'
    return x -> \st -> return x
    finally f -> f v
  } as x in return x
```

# Miro - Using instances

## Example

```
postInc : forall s. Inst s State -> Int!{s}
postInc [s] inst =
  x <- inst#get();
  inst#put(x + 1);
  return x
```

# Miro - Running instances

## Example

```
result : Int!{}  
result =  
  runscope(s1 ->  
    r1 <- ref [s1] 10;  
    runscope(s2 ->  
      r2 <- ref [s2] 20;  
      x <- postInc [s2] r2;  
      r1#put(x);  
      return x);  
    y <- r1#get ();  
    return y) -- result is 20
```

# Next Subsection

- 1 What are effects
- 2 Algebraic effects and handlers
- 3 Miro
- 4 Semantics
- 5 Type system  
Type safety