Tracer – A reimplementation of Stepper

Supplementary slides

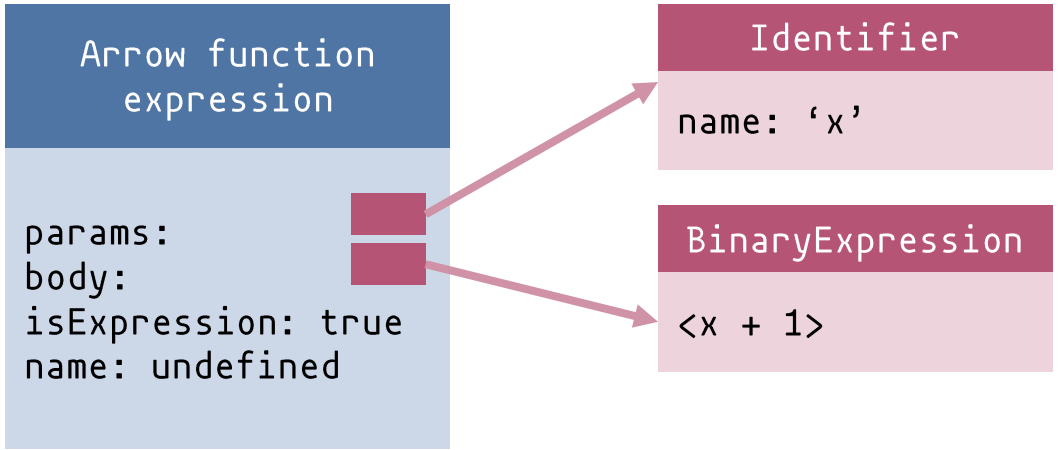
Prepared by **CATISNOTSODIUM** 😺

Functions and mu terms

Functions

How are functions represented in Stepper?

```
x => x + 1
```



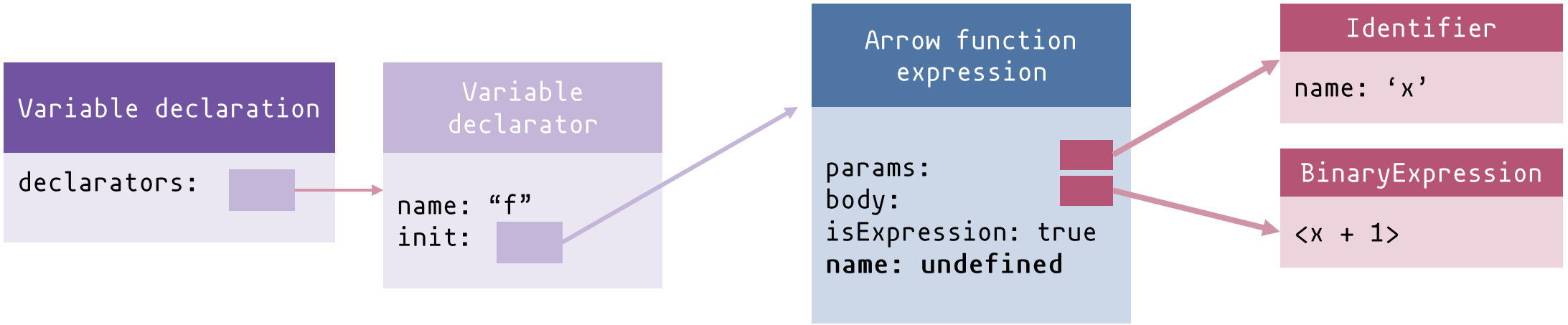
<*> details are omitted

Functions

How are functions represented in Stepper?

```
const f = x => x + 1;
```

AST of f

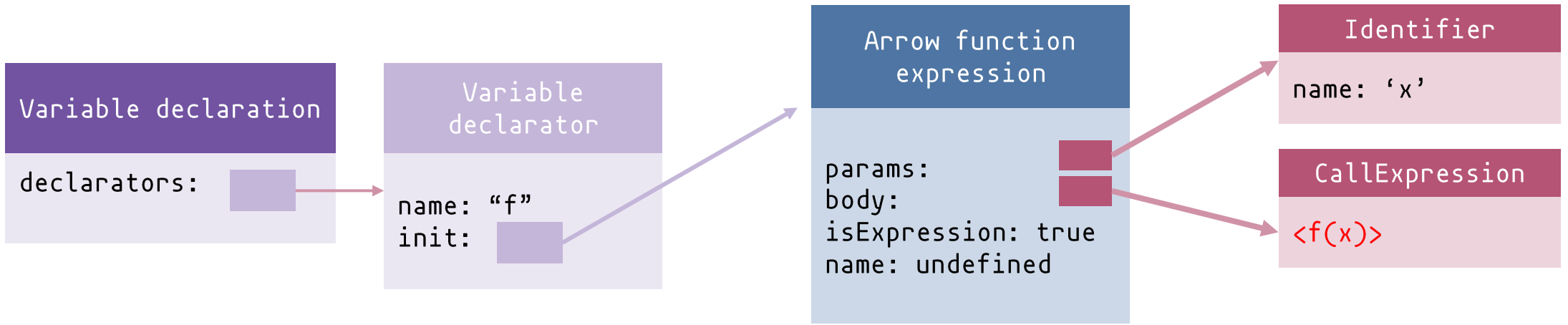


Functions

How are functions represented in Stepper?

```
const f = x => f(x);
```

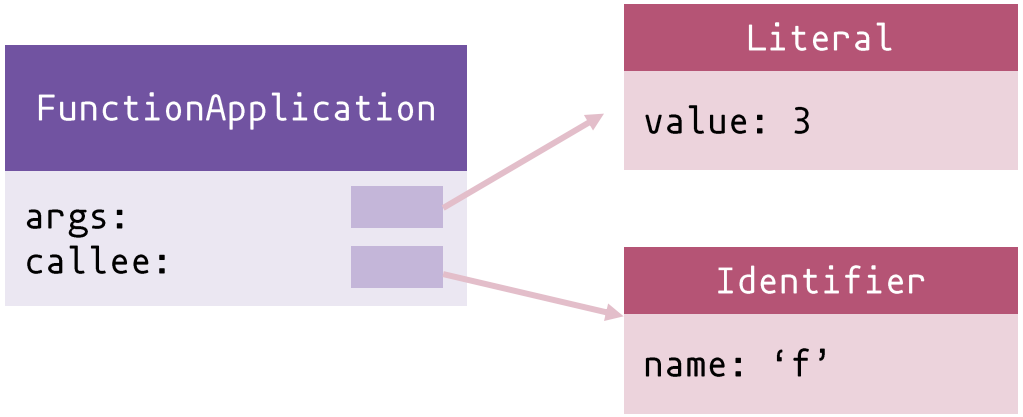
AST of f



Functions

How are functions represented in Stepper?

```
const f = x => f(x);
f(3);
```



Functions: Evaluation

```
const f = x => f(x);
f(3);
```

Next step

```
(x => (x => f(x))(x))(3);
```

After substitution of f, all occurrences of f in the body will be substituted with `x => f(x)`.

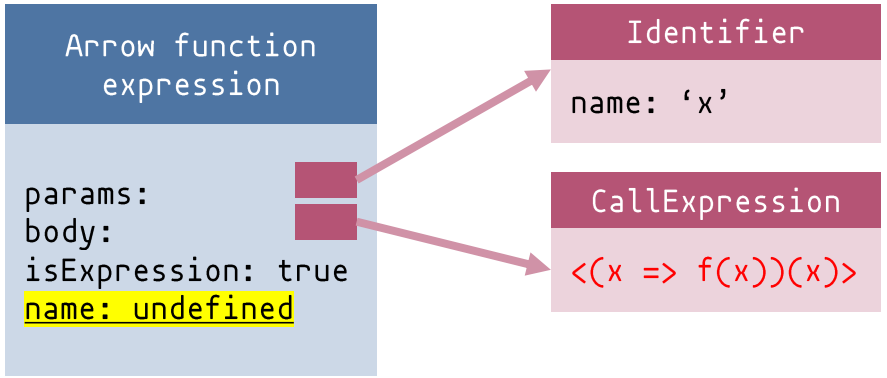
Functions: Evaluation

```
const f = x => f(x);
f(3);
```

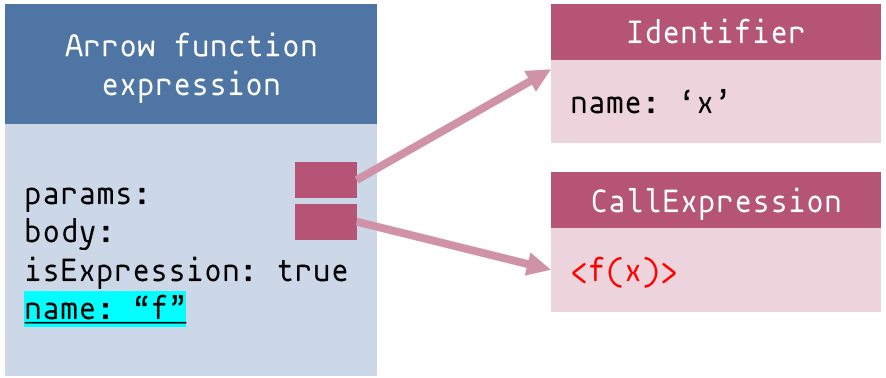
Next step

```
(x => (x => f(x))(x))(3);
```

`(x => (x => f(x))(x))`



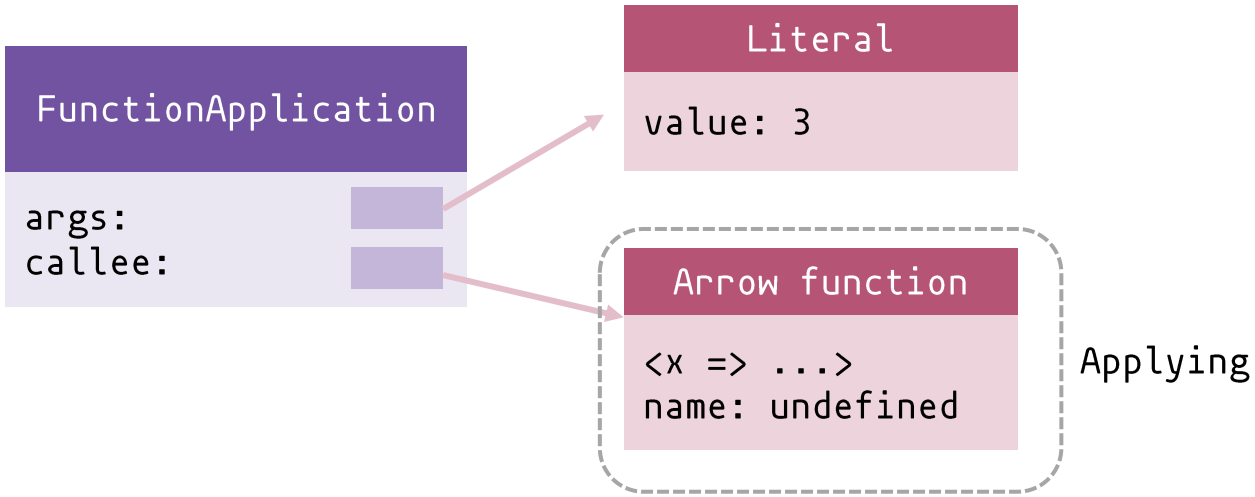
`x => f(x)`



After substitution of f, all occurrences of f in the body will be substituted with `x => f(x)`.

Functions: Evaluation

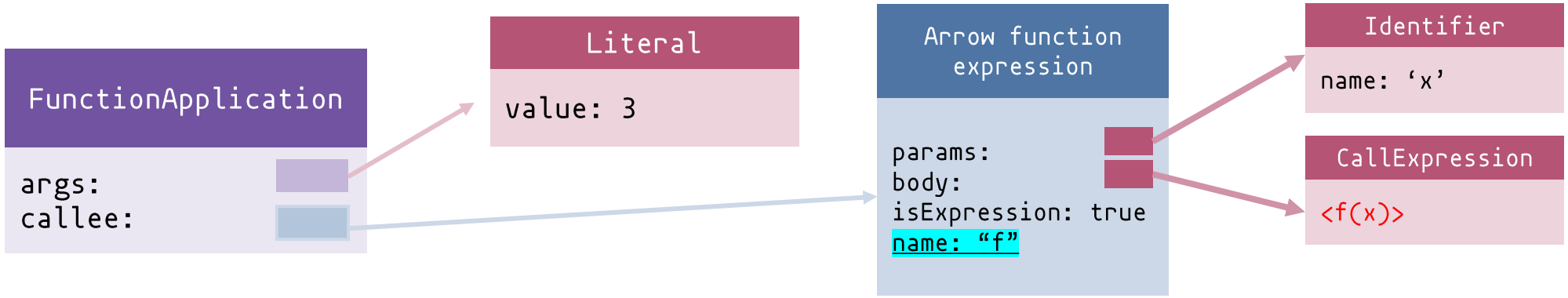
```
(x => (x => f(x))(x))(3);
```



Function application

Functions: Evaluation

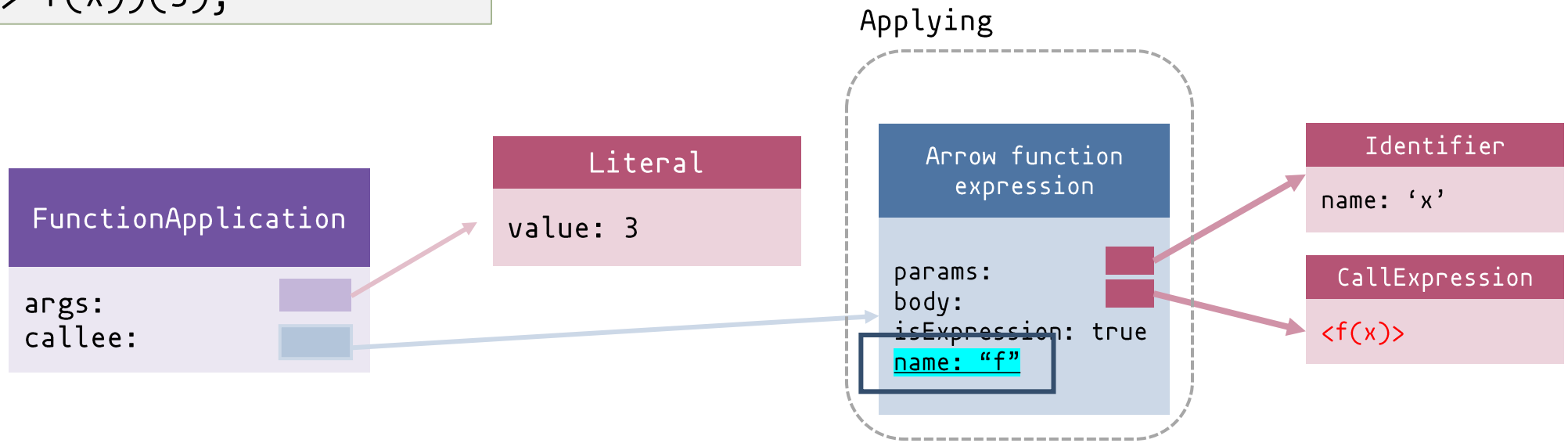
```
(x => f(x))(3);
```



Function application

Functions: Evaluation

```
(x => f(x))(3);
```



Since function x => f(x) has a name "f", all occurrences of f in the body will be substituted with x => f(x).

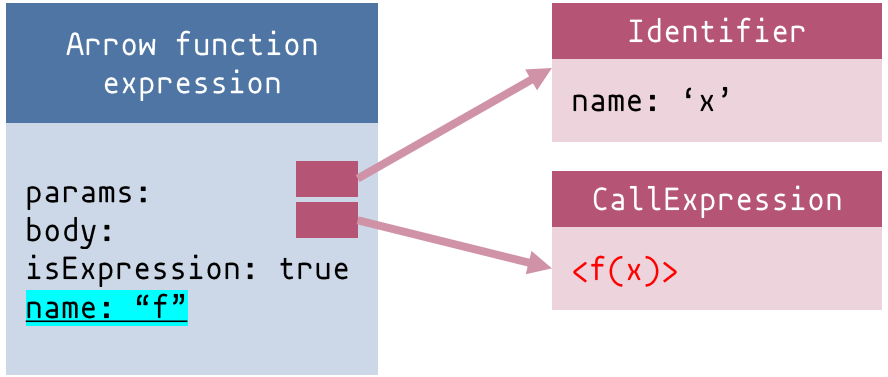
Functions: Evaluation

Since function x => f(x) has a name “f”, all occurrences of f in the body will be substituted with x => f(x).

```
(x => f(x))(3);
```

```
(x => (x => f(x))(x))(3);
```

```
(x => f(x))(3);
```



Functions: Evaluation

```
const f = x => f(x);
f(3);
```

```
(x => (x => f(x))(x))(3);
```

```
(x => f(x))(3);
```

Extra step:
substituting f

```
(x => (x => f(x))(x))(3);
```

```
(x => f(x))(3);
```

Functions: Evaluation

```
const f = x => f(x);
f(3);
```

```
(x => (x => f(x))(x))(3);
```

```
(x => f(x))(3);
```

```
(x => (x => f(x))(x))(3);
```

```
(x => f(x))(3);
```

The substitution still works even though f has already been substituted.