Degrees of Relatedness

Andreas Nuyts

KU Leuven, Belgium

EUTypes 2018 Working Meeting
Nijmegen, Netherlands
January 24, 2018

One Slide Summary

- Parametricity is about **relations**,
- Objects are **related** $\rightsquigarrow$ Specify to what **degree** i $(s \frown_i t)$,
- The **larger** the type, the **more degrees** are eligible,
- Describe **function behaviour** by saying how functions **influence degree** of relatedness,
- This explains
  - **parametricity:** $\text{flatten} : (\textbf{par} \mid X : \mathcal{U}) \to \text{Tree}\ X \to \text{List}\ X$
  - **ad hoc polymorphism:** $\text{lem} : (\textbf{hoc} \mid X : \mathcal{U}) \to X \uplus (X \to \text{Empty})$
  - **. irrelevance:** $[] : (\textbf{irr} \mid n : \mathbb{N}) \to \text{List}_n\ A$
  - **.. shape-irrelevance:** $\lambda n.\text{List}_n\ A : (\textbf{shi} \mid n : \mathbb{N}) \to \mathcal{U}$
  - aspects of **unions, intersections, algebra, Prop, . . .**

One Slide Summary

- Parametricity is about **relations**,
- Objects are **related** $\leadsto$ Specify to what **degree** i ($s \frown_i t$),
- The **larger** the type, the **more degrees** are eligible,
- Describe **function behaviour** by saying how functions **influence degree** of relatedness,
- This explains
  - **parametricity:** $\text{flatten} : (\textbf{par} \mid X : \mathcal{U}) \to \text{Tree}\ X \to \text{List}\ X$
  - **ad hoc polymorphism:** $\text{lem} : (\textbf{hoc} \mid X : \mathcal{U}) \to X \uplus (X \to \text{Empty})$
  - **. irrelevance:** $[] : (\textbf{irr} \mid n : \mathbb{N}) \to \text{List}_n\ A$
  - **.. shape-irrelevance:** $\lambda n.\text{List}_n\ A : (\textbf{shi} \mid n : \mathbb{N}) \to \mathcal{U}$
  - aspects of **unions, intersections, algebra, Prop, …**

One Slide Summary

- Parametricity is about **relations**,
- Objects are **related** $\rightsquigarrow$ Specify to what **degree** i ($s \frown_i t$),
- The **larger** the type, the **more degrees** are eligible,
- Describe **function behaviour** by saying how functions **influence degree** of relatedness,
- This explains
  - **parametricity:** $\mathrm{flatten} : (\mathbf{par} \mid X : \mathcal{U}) \to \mathrm{Tree}\ X \to \mathrm{List}\ X$
  - **ad hoc polymorphism:** $\mathrm{lem} : (\mathbf{hoc} \mid X : \mathcal{U}) \to X \uplus (X \to \mathrm{Empty})$
  - **. irrelevance:** $[] : (\mathbf{irr} \mid n : \mathbb{N}) \to \mathrm{List}_n\ A$
  - **.. shape-irrelevance:** $\lambda n.\mathrm{List}_n\ A : (\mathbf{shi} \mid n : \mathbb{N}) \to \mathcal{U}$
  - aspects of **unions, intersections, algebra, Prop, . . .**

One Slide Summary

- Parametricity is about **relations**,
- Objects are **related** $\rightsquigarrow$ Specify to what **degree** i ($s \frown_i t$),
- The **larger** the type, the **more degrees** are eligible,
- Describe **function behaviour** by saying how functions **influence degree** of relatedness,
- This explains
  - **parametricity:** $\mathsf{flatten} : (\mathbf{par} \mid X : \mathcal{U}) \to \mathsf{Tree}\ X \to \mathsf{List}\ X$
  - **ad hoc polymorphism:** $\mathsf{lem} : (\mathbf{hoc} \mid X : \mathcal{U}) \to X \uplus (X \to \mathsf{Empty})$
  - **. irrelevance:** $[] : (\mathbf{irr} \mid n : \mathbb{N}) \to \mathsf{List}_n\ A$
  - **.. shape-irrelevance:** $\lambda n.\mathsf{List}_n\ A : (\mathbf{shi} \mid n : \mathbb{N}) \to \mathcal{U}$
  - aspects of **unions, intersections, algebra, Prop, . . .**

One Slide Summary

- Parametricity is about **relations**,
- Objects are **related** $\rightsquigarrow$ Specify to what **degree** i ($s \frown_i t$),
- The **larger** the type, the **more degrees** are eligible,
- Describe **function behaviour** by saying how functions **influence degree** of relatedness,
- This explains
  - **parametricity:** flatten : $(\mathbf{par} \mid X : \mathcal{U}) \to \text{Tree } X \to \text{List } X$
  - **ad hoc polymorphism:** lem : $(\mathbf{hoc} \mid X : \mathcal{U}) \to X \uplus (X \to \text{Empty})$
  - **. irrelevance:** [] : $(\mathbf{irr} \mid n : \mathbb{N}) \to \text{List}_n\, A$
  - **.. shape-irrelevance:** $\lambda n.\text{List}_n\, A : (\mathbf{shi} \mid n : \mathbb{N}) \to \mathcal{U}$
  - aspects of **unions, intersections, algebra, Prop, . . .**

One Slide Summary

- Parametricity is about **relations**,
- Objects are **related** $\rightsquigarrow$ Specify to what **degree** i ($s \frown_i t$),
- The **larger** the type, the **more degrees** are eligible,
- Describe **function behaviour** by saying how functions **influence degree** of relatedness,
- This explains
  - **parametricity:** flatten : (**par** $\shortmid X : \mathcal{U}) \to \text{Tree } X \to \text{List } X$
  - **ad hoc polymorphism:** lem : (**hoc** $\shortmid X : \mathcal{U}) \to X \uplus (X \to \text{Empty})$
  - . **irrelevance:** $[] : (\textbf{irr} \shortmid n : \mathbb{N}) \to \text{List}_n\, A$
  - .. **shape-irrelevance:** $\lambda n.\text{List}_n\, A : (\textbf{shi} \shortmid n : \mathbb{N}) \to \mathcal{U}$
  - aspects of **unions, intersections, algebra, Prop, . . .**

One Slide Summary

- Parametricity is about **relations**,
- Objects are **related** $\rightsquigarrow$ Specify to what **degree** i ($s \frown_i t$),
- The **larger** the type, the **more degrees** are eligible,
- Describe **function behaviour** by saying how functions **influence degree** of relatedness,
- This explains
  - **parametricity:** flatten : $(\mathbf{par} \mid X : \mathcal{U}) \to \text{Tree } X \to \text{List } X$
  - **ad hoc polymorphism:** lem : $(\mathbf{hoc} \mid X : \mathcal{U}) \to X \uplus (X \to \text{Empty})$
  - **. irrelevance:** $[]$: $(\mathbf{irr} \mid n : \mathbb{N}) \to \text{List}_n\ A$
  - **.. shape-irrelevance:** $\lambda n.\text{List}_n\ A : (\mathbf{shi} \mid n : \mathbb{N}) \to \mathcal{U}$
  - aspects of **unions, intersections, algebra, Prop, . . .**

One Slide Summary

- Parametricity is about **relations**,
- Objects are **related** $\leadsto$ Specify to what **degree** i ($s \frown_i t$),
- The **larger** the type, the **more degrees** are eligible,
- Describe **function behaviour** by saying how functions **influence degree** of relatedness,
- This explains
  - **parametricity:** flatten : $(\textbf{par} \shortmid X : \mathcal{U}) \to \text{Tree } X \to \text{List } X$
  - **ad hoc polymorphism:** lem : $(\textbf{hoc} \shortmid X : \mathcal{U}) \to X \uplus (X \to \text{Empty})$
  - **. irrelevance:** [] : $(\textbf{irr} \shortmid n : \mathbb{N}) \to \text{List}_n\ A$
  - **.. shape-irrelevance:** $\lambda n.\text{List}_n\ A : (\textbf{shi} \shortmid n : \mathbb{N}) \to \mathcal{U}$
  - aspects of **unions, intersections, algebra, Prop, . . .**

One Slide Summary

- Parametricity is about **relations**,
- Objects are **related** $\rightsquigarrow$ Specify to what **degree** i ($s \frown_i t$),
- The **larger** the type, the **more degrees** are eligible,
- Describe **function behaviour** by saying how functions **influence degree** of relatedness,
- This explains
  - **parametricity:** flatten : $(\mathbf{par} \shortmid X : \mathcal{U}) \to \text{Tree } X \to \text{List } X$
  - **ad hoc polymorphism:** lem : $(\mathbf{hoc} \shortmid X : \mathcal{U}) \to X \uplus (X \to \text{Empty})$
  - **. irrelevance:** [] : $(\mathbf{irr} \shortmid n : \mathbb{N}) \to \text{List}_n\ A$
  - **.. shape-irrelevance:** $\lambda n.\text{List}_n\ A : (\mathbf{shi} \shortmid n : \mathbb{N}) \to \mathcal{U}$
  - aspects of **unions, intersections, algebra, Prop, . . .**

Theorem

$$(A \to B) \cong \left(\underbrace{\forall X.(X \to A)}_{\textit{For any representation } (X, r) \textit{ of } A} \to (X \to B) \right)$$

Proof:

($\to$) $h \mapsto \lambda X.\lambda r.h \circ r$.

($\leftarrow$) $g \mapsto g\ A\ \mathrm{id}$.

(src) refl

(tgt) Prove: $g\ X\ r\ x = g\ A\ \mathrm{id}\ (r\ x)$.

Parametricity: Representation independence

Theorem

$$(A \to B) \cong \left(\underbrace{\forall X.(X \to A)}_{\text{For any representation } (X, r) \text{ of } A} \to (X \to B) \right)$$

Proof:

(→) $h \mapsto \lambda X.\lambda r.h \circ r$.

(←) $g \mapsto g\ A\ \mathrm{id}$.

(src) refl

(tgt) Prove: $g\ X\ r\ x = g\ A\ \mathrm{id}\ (r\ x)$.

Parametricity: Representation independence

Theorem

$$(A \to B) \cong \left(\underbrace{\forall X.(X \to A)}_{\text{For any representation } (X, r) \text{ of } A} \to (X \to B) \right)$$

Proof:

$(\to)$ $h \mapsto \lambda X.\lambda r.h \circ r.$

$(\leftarrow)$ $g \mapsto g\ A\ \mathrm{id}.$

(src) refl

(tgt) Prove: $g\ X\ r\ x = g\ A\ \mathrm{id}\ (r\ x).$

Parametricity: Representation independence

Theorem

$$(A \to B) \cong \left(\underbrace{\forall X.(X \to A)}_{\textit{For any representation } (X, r) \textit{ of } A} \to (X \to B) \right)$$

Proof:

($\to$) $h \mapsto \lambda X.\lambda r.h \circ r$.

($\leftarrow$) $g \mapsto g\ A\ \mathrm{id}$.

(src) refl

(tgt) Prove: $g\ X\ r\ x = g\ A\ \mathrm{id}\ (r\ x)$.

Parametricity: Representation independence

Theorem

$$(A \to B) \cong \left(\underbrace{\forall X.(X \to A)}_{\text{For any representation } (X, r) \text{ of } A} \to (X \to B) \right)$$

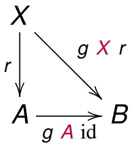
Proof:

($\to$) $h \mapsto \lambda X.\lambda r.h \circ r$.

($\leftarrow$) $g \mapsto g\ A\ \mathrm{id}$.

(src) refl

(tgt) Prove: $g\ X\ r\ x = g\ A\ \mathrm{id}\ (r\ x)$.



Relational parametricity

Lemma

If $g : \forall X.(X \to A) \to (X \to B)$
then $g\ X_0\ r_0\ x_0 = g\ A\ \mathrm{id}\ (r_0\ x_0)$.

Rel. param.: A sound scheme for proving parametricity theorems.
Idea: **Related things map to related things.**

Lemma

If $g : \forall X.(X \to A) \to (X \to B)$
then $g\ X_0\ r_0\ x_0 = g\ A\ \mathrm{id}\ (r_0\ x_0)$.

Rel. param.: A sound scheme for proving parametricity theorems.
Idea: **Related things map to related things.**

Lemma

If $g : \forall X.(X \to A) \to (X \to B)$
then $g\ X_0\ r_0\ x_0 = g\ A\ \mathrm{id}\ (r_0\ x_0)$.

Rel. param.: A sound scheme for proving parametricity theorems.
Idea: **Related things map to related things.**

$$X : *, \qquad r : X \to A, \qquad x : X \qquad \vdash \qquad g\ X\ r\ x : B$$

Relational parametricity

Lemma

If $g : \forall X.(X \to A) \to (X \to B)$
then $g\ X_0\ r_0\ x_0 = g\ A\ \mathrm{id}\ (r_0\ x_0)$.

Rel. param.: A sound scheme for proving parametricity theorems.
Idea: **Related things map to related things.**

$$
\begin{array}{lllll}
X_0 : *, & r_0 : X_0 \to A, & x_0 : X_0 & \vdash & g\ X_0\ r_0\ x_0 : B \\
[X] : \mathrm{Rel}, & [r] : [X \to A], & [x] : [X] & \vdash & \\
X_1 : *, & r_1 : X_1 \to A, & x_1 : X_1 & \vdash & g\ X_1\ r_1\ x_1 : B
\end{array}
$$

Relational parametricity

Lemma

If $g : \forall X.(X \to A) \to (X \to B)$
then $g\ X_0\ r_0\ x_0 = g\ A\ \mathrm{id}\ (r_0\ x_0)$.

Rel. param.: A sound scheme for proving parametricity theorems.
Idea: **Related things map to related things.**

$$
\begin{array}{lllll}
X_0 : *, & r_0 : X_0 \to A, & x_0 : X_0 & \vdash & g\ X_0\ r_0\ x_0 : B \\
[X] : \mathrm{Rel}, & [r] : [X \to A], & [x] : [X] & \vdash & \\
X_1 : *, & r_1 : X_1 \to A, & x_1 : X_1 & \vdash & g\ X_1\ r_1\ x_1 : B
\end{array}
$$

Relational parametricity

Lemma

If $g : \forall X.(X \to A) \to (X \to B)$
then $g\ X_0\ r_0\ x_0 = g\ A\ \mathrm{id}\ (r_0\ x_0)$.

Rel. param.: A sound scheme for proving parametricity theorems.
Idea: **Related things map to related things.**

$$
\begin{array}{lllll}
X_0 : *, & r_0 : X_0 \to A, & x_0 : X_0 & \vdash & g\ X_0\ r_0\ x_0 : B \\
[X] : \mathrm{Rel}, & [r] : [X \to A], & [x] : [X] & \vdash & \\
X_1 : *, & r_1 : X_1 \to A, & x_1 : X_1 & \vdash & g\ X_1\ r_1\ x_1 : B
\end{array}
$$

Lemma

If $g : \forall X.(X \to A) \to (X \to B)$
then $g\ X_0\ r_0\ x_0 = g\ A\ \mathrm{id}\ (r_0\ x_0)$.

Rel. param.: A sound scheme for proving parametricity theorems.
Idea: **Related things map to related things.**

$$
\begin{array}{lllcl}
X_0 : *, & r_0 : X_0 \to A, & x_0 : X_0 & \vdash & g\ X_0\ r_0\ x_0 : B \\
[X] : \mathrm{Rel}, & [r] : [X \to A], & [x] : [X] & \vdash & \\
X_1 : *, & r_1 : X_1 \to A, & x_1 : X_1 & \vdash & g\ X_1\ r_1\ x_1 : B
\end{array}
$$

Lemma

If $g : \forall X.(X \to A) \to (X \to B)$
then $g\ X_0\ r_0\ x_0 = g\ A\ \mathrm{id}\ (r_0\ x_0)$.

Rel. param.: A sound scheme for proving parametricity theorems.
Idea: **Related things map to related things.**

$$
\begin{array}{ccccc}
X_0 : *, & r_0 : X_0 \to A, & x_0 : X_0 & \vdash & g\ X_0\ r_0\ x_0 : B \\
[X] : \mathrm{Rel}, & [r] : [X \to A], & [x] : [X] & \vdash & [g\ X\ r\ x] : [B] \\
X_1 : *, & r_1 : X_1 \to A, & x_1 : X_1 & \vdash & g\ X_1\ r_1\ x_1 : B
\end{array}
$$

Lemma

If $g : \forall X.(X \to A) \to (X \to B)$
then $g\ X_0\ r_0\ x_0 = g\ A\ \mathrm{id}\ (r_0\ x_0)$.

Rel. param.: A sound scheme for proving parametricity theorems.
Idea: **Related things map to related things.**

$$
\begin{array}{ccccc}
X_0 : *, & r_0 : X_0 \to A, & x_0 : X_0 & \vdash & g\ X_0\ r_0\ x_0 : B \\
[X] : \mathrm{Rel}, & [r] : [X \to A], & [x] : [X] & \vdash & = \\
X_1 : *, & r_1 : X_1 \to A, & x_1 : X_1 & \vdash & g\ X_1\ r_1\ x_1 : B
\end{array}
$$

IDENTITY EXTENSION LEMMA (IEL)

Lemma

If $g : \forall X.(X \to A) \to (X \to B)$
then $g\ X_0\ r_0\ x_0 = g\ A\ \mathrm{id}\ (r_0\ x_0)$.

Rel. param.: A sound scheme for proving parametricity theorems.
Idea: **Related things map to related things.**

$X_0 : *,$	$r_0 : X_0 \to A,$	$x_0 : X_0$	$\vdash$	$g\ X_0\ r_0\ x_0 : B$
$[X] : \mathrm{Rel},$	$[r] : [X \to A],$	$[x] : [X]$	$\vdash$	=
$A : *,$	$\mathrm{id} : A \to A,$	$r_0\ x_0 : A$	$\vdash$	$g\ A\ \mathrm{id}\ (r_0\ x_0) : B$

IDENTITY EXTENSION LEMMA (IEL)

Lemma

If $g : \forall X.(X \to A) \to (X \to B)$
then $g\ X_0\ r_0\ x_0 = g\ A\ \mathrm{id}\ (r_0\ x_0)$.

Rel. param.: A sound scheme for proving parametricity theorems.
Idea: **Related things map to related things.**

$$\begin{array}{ccccc} X_0 : *, & r_0 : X_0 \to A, & x_0 : X_0 & \vdash & g\ X_0\ r_0\ x_0 : B \\ /r_0/ : \mathrm{Rel}, & [r] : \backslash \sqcup \circ r_0 \backslash, & [x] : /r_0/ & \vdash & = \\ A : *, & \mathrm{id} : A \to A, & r_0\ x_0 : A & \vdash & g\ A\ \mathrm{id}\ (r_0\ x_0) : B \end{array}$$

IDENTITY EXTENSION LEMMA (IEL)

Lemma

If $g : \forall X.(X \to A) \to (X \to B)$
then $g\ X_0\ r_0\ x_0 = g\ A\ \mathrm{id}\ (r_0\ x_0)$.

Rel. param.: A sound scheme for proving parametricity theorems.
Idea: **Related things map to related things.**

$X_0 : *,$	$r_0 : X_0 \to A,$	$x_0 : X_0$	$\vdash$	$g\ X_0\ r_0\ x_0 : B$
$/r_0/ : \mathrm{Rel},$	refl : $\backslash \sqcup \circ r_0 \backslash,$	refl : $/r_0/$	$\vdash$	**=**
$A : *,$	$\mathrm{id} : A \to A,$	$r_0\ x_0 : A$	$\vdash$	$g\ A\ \mathrm{id}\ (r_0\ x_0) : B$

IDENTITY EXTENSION LEMMA (IEL)

Π is not parametric

System F:
$\forall X.(X \to A) \to (X \to B)$.

Dependent types:
$\Pi(X : \mathcal{U}).(X \to A) \to (X \to B)$.

Suppose $B = \mathcal{U}$:
$leak : \Pi(X : \mathcal{U}).(X \to A) \to (X \to \mathcal{U})$
$leak\ X\ r\ x = X$.
Representation type is returned as data!

But think of $Leak\ X\ r\ x = X$ as a dependent type
We're just ignoring arguments

DTT: formal type/data boundary disappears
Difference in expectation remains

Π is not parametric

System F:
$\forall X.(X \to A) \to (X \to B)$.

Dependent types:
$\Pi(X : \mathcal{U}).(X \to A) \to (X \to B)$.

Suppose $B = \mathcal{U}$:
$leak : \Pi(X : \mathcal{U}).(X \to A) \to (X \to \mathcal{U})$
$leak\ X\ r\ x = X$.
Representation type is returned as data!

But think of $Leak\ X\ r\ x = X$ as a dependent type
We're just ignoring arguments

DTT: formal type/data boundary disappears
Difference in expectation remains

Π is not parametric

System F:
$\forall X.(X \to A) \to (X \to B)$.

Dependent types:
$\Pi(X : \mathcal{U}).(X \to A) \to (X \to B)$.

Suppose $B = \mathcal{U}$:
$\mathit{leak} : \Pi(X : \mathcal{U}).(X \to A) \to (X \to \mathcal{U})$
$\mathit{leak}\ X\ r\ x = X$.
Representation type is returned as data!

But think of $\mathit{Leak}\ X\ r\ x = X$ as a dependent type
We're just ignoring arguments

DTT: formal type/data boundary disappears
Difference in expectation remains

Π is not parametric

System F:
$\forall X.(X \to A) \to (X \to B)$.

Dependent types:
$\Pi(X : \mathcal{U}).(X \to A) \to (X \to B)$.

Suppose $B = \mathcal{U}$:
$\mathit{leak} : \Pi(X : \mathcal{U}).(X \to A) \to (X \to \mathcal{U})$
$\mathit{leak}\ X\ r\ x = X$.
Representation type is returned as data!

But think of $\mathit{Leak}\ X\ r\ x = X$ as a dependent type
We're just ignoring arguments

DTT: formal type/data boundary disappears
Difference in expectation remains

Π is not parametric

System F:
$\forall X.(X \to A) \to (X \to B)$.

Dependent types:
$\Pi(X : \mathcal{U}).(X \to A) \to (X \to B)$.

Suppose $B = \mathcal{U}$:
$leak : \Pi(X : \mathcal{U}).(X \to A) \to (X \to \mathcal{U})$
$leak\ X\ r\ x = X$.
Representation type is returned as data!

But think of $Leak\ X\ r\ x = X$ as a dependent type
We're just ignoring arguments

DTT: formal type/data boundary disappears
Difference in expectation remains

Let's have two relations

System F

Values can be related:

$$(s : S) \frown (t : T)$$

IEL: if $(s : A) \frown (t : A)$ then $s = t$ (heterogeneous equality)

Types can be related:

$$S \frown T$$

which gives meaning to

$$(s : S) \frown (t : T)$$

Dependent types

Things can be 0-related:

$$(s : S) \frown_0 (t : T)$$

IEL: if $(s : A) \frown_0 (t : A)$ then $s = t$ (heterogeneous equality)

Things can be 1-related:

$$(s : K) \frown_1 (t : L)$$

where $(S : \mathcal{U}) \frown_1 (T : \mathcal{U})$ gives meaning to

$$(s : S) \frown_0 (t : T)$$

Let's have two relations

System F

Values can be related:

$$(s : S) \frown (t : T)$$

IEL: if $(s : A) \frown (t : A)$ then $s = t$ (heterogeneous equality)

Types can be related:

$$S \frown T$$

which gives meaning to

$$(s : S) \frown (t : T)$$

Dependent types

Things can be 0-related:

$$(s : S) \frown_0 (t : T)$$

IEL: if $(s : A) \frown_0 (t : A)$ then $s = t$ (heterogeneous equality)

Things can be 1-related:

$$(s : K) \frown_1 (t : L)$$

where $(S : \mathcal{U}) \frown_1 (T : \mathcal{U})$ gives meaning to

$$(s : S) \frown_0 (t : T)$$

Let's have two relations

System F

Values can be related:

$$(s : S) \frown (t : T)$$

IEL: if $(s : A) \frown (t : A)$ then $s = t$ (heterogeneous equality)

Types can be related:

$$S \frown T$$

which gives meaning to

$$(s : S) \frown (t : T)$$

Dependent types

Things can be 0-related:

$$(s : S) \frown_0 (t : T)$$

IEL: if $(s : A) \frown_0 (t : A)$ then $s = t$ (heterogeneous equality)

Things can be 1-related:

$$(s : K) \frown_1 (t : L)$$

where $(S : \mathcal{U}) \frown_1 (T : \mathcal{U})$ gives meaning to

$$(s : S) \frown_0 (t : T)$$

System F

Values can be related:

$$(s : S) \frown (t : T)$$

IEL: if $(s : A) \frown (t : A)$ then $s = t$ (heterogeneous equality)

Types can be related:

$$S \frown T$$

which gives meaning to

$$(s : S) \frown (t : T)$$

Dependent types

Things can be 0-related:

$$(s : S) \frown_0 (t : T)$$

IEL: if $(s : A) \frown_0 (t : A)$ then $s = t$ (heterogeneous equality)

Things can be 1-related:

$$(s : K) \frown_1 (t : L)$$

where $(S : \mathcal{U}) \frown_1 (T : \mathcal{U})$ gives meaning to

$$(s : S) \frown_0 (t : T)$$

Examples

0-relatedness:

- $(2+5 : \mathbb{N}) \frown_0 (7 : \mathbb{N})$
- $([\text{true}, \text{false}] : \text{List}_4\ \text{Bool}) \frown_0 ([\text{true}, \text{false}] : \text{List}_7\ \text{Bool})$
- $(\text{flatten Bool} : \text{Tree Bool} \to \text{List Bool}) \frown_0$
  $(\text{flatten}\ \mathbb{N} : \text{Tree}\ \mathbb{N} \to \text{List}\ \mathbb{N})$
  for any proof of $\text{Bool} \frown_1 \mathbb{N}$

1-relatedness:

- $(\text{List}_4\ \text{Bool} : \mathcal{U}) \frown_1 (\text{List}_7\ \text{Bool} : \mathcal{U})$
- $(\text{Bool} : \mathcal{U}) \frown_1 (\mathbb{N} : \mathcal{U})$ non-canonically
  (e.g. by setting $\text{true} \frown_0 5$ and $\text{false} \frown_0 2k+1$)

2-relatedness:

- $(\text{Monoid} : \mathcal{U}) \frown_2 (\text{Group} : \mathcal{U})$
  by setting $(M : \text{Monoid}) \frown_1 (G : \text{Group})$
  whenever $(M : \text{Monoid}) \frown_1 (\text{asMonoid}\ G : \text{Monoid})$.

...

Examples

0-relatedness:

- $(2 + 5 : \mathbb{N}) \frown_0 (7 : \mathbb{N})$
- $([\mathsf{true}, \mathsf{false}] : \mathsf{List}_4\ \mathsf{Bool}) \frown_0 ([\mathsf{true}, \mathsf{false}] : \mathsf{List}_7\ \mathsf{Bool})$
- $(\mathsf{flatten}\ \mathsf{Bool} : \mathsf{Tree}\ \mathsf{Bool} \to \mathsf{List}\ \mathsf{Bool}) \frown_0$
  $(\mathsf{flatten}\ \mathbb{N} : \mathsf{Tree}\ \mathbb{N} \to \mathsf{List}\ \mathbb{N})$
  for any proof of $\mathsf{Bool} \frown_1 \mathbb{N}$

1-relatedness:

- $(\mathsf{List}_4\ \mathsf{Bool} : \mathcal{U}) \frown_1 (\mathsf{List}_7\ \mathsf{Bool} : \mathcal{U})$
- $(\mathsf{Bool} : \mathcal{U}) \frown_1 (\mathbb{N} : \mathcal{U})$ non-canonically
  (e.g. by setting $\mathsf{true} \frown_0 5$ and $\mathsf{false} \frown_0 2k + 1$)

2-relatedness:

- $(\mathsf{Monoid} : \mathcal{U}) \frown_2 (\mathsf{Group} : \mathcal{U})$
  by setting $(M : \mathsf{Monoid}) \frown_1 (G : \mathsf{Group})$
  whenever $(M : \mathsf{Monoid}) \frown_1 (\mathsf{asMonoid}\ G : \mathsf{Monoid})$.

...

Examples

0-relatedness:

- $(2 + 5 : \mathbb{N}) \frown_0 (7 : \mathbb{N})$
- $([\mathsf{true}, \mathsf{false}] : \mathsf{List}_4\ \mathsf{Bool}) \frown_0 ([\mathsf{true}, \mathsf{false}] : \mathsf{List}_7\ \mathsf{Bool})$
- $(\mathsf{flatten}\ \mathsf{Bool} : \mathsf{Tree}\ \mathsf{Bool} \to \mathsf{List}\ \mathsf{Bool}) \frown_0$
  $(\mathsf{flatten}\ \mathbb{N} : \mathsf{Tree}\ \mathbb{N} \to \mathsf{List}\ \mathbb{N})$
  for any proof of $\mathsf{Bool} \frown_1 \mathbb{N}$

1-relatedness:

- $(\mathsf{List}_4\ \mathsf{Bool} : \mathcal{U}) \frown_1 (\mathsf{List}_7\ \mathsf{Bool} : \mathcal{U})$
- $(\mathsf{Bool} : \mathcal{U}) \frown_1 (\mathbb{N} : \mathcal{U})$ non-canonically
  (e.g. by setting $\mathsf{true} \frown_0 5$ and $\mathsf{false} \frown_0 2k + 1$)

2-relatedness:

- $(\mathsf{Monoid} : \mathcal{U}) \frown_2 (\mathsf{Group} : \mathcal{U})$
  by setting $(M : \mathsf{Monoid}) \frown_1 (G : \mathsf{Group})$
  whenever $(M : \mathsf{Monoid}) \frown_1 (\mathsf{asMonoid}\ G : \mathsf{Monoid})$.

...

Examples

0-relatedness:

- $(2+5 : \mathbb{N}) \frown_0 (7 : \mathbb{N})$
- $([\text{true}, \text{false}] : \text{List}_4\ \text{Bool}) \frown_0 ([\text{true}, \text{false}] : \text{List}_7\ \text{Bool})$
- $(\text{flatten Bool} : \text{Tree Bool} \to \text{List Bool}) \frown_0$
  $(\text{flatten}\ \mathbb{N} : \text{Tree}\ \mathbb{N} \to \text{List}\ \mathbb{N})$
  for any proof of $\text{Bool} \frown_1 \mathbb{N}$

1-relatedness:

- $(\text{List}_4\ \text{Bool} : \mathcal{U}) \frown_1 (\text{List}_7\ \text{Bool} : \mathcal{U})$
- $(\text{Bool} : \mathcal{U}) \frown_1 (\mathbb{N} : \mathcal{U})$ non-canonically
  (e.g. by setting $\text{true} \frown_0 5$ and $\text{false} \frown_0 2k+1$)

2-relatedness:

- $(\text{Monoid} : \mathcal{U}) \frown_2 (\text{Group} : \mathcal{U})$
  by setting $(M : \text{Monoid}) \frown_1 (G : \text{Group})$
  whenever $(M : \text{Monoid}) \frown_1 (\text{asMonoid}\ G : \text{Monoid})$.

...

Examples

0-relatedness:

- $(2+5 : \mathbb{N}) \frown_0 (7 : \mathbb{N})$
- $([\mathsf{true}, \mathsf{false}] : \mathsf{List}_4\ \mathsf{Bool}) \frown_0 ([\mathsf{true}, \mathsf{false}] : \mathsf{List}_7\ \mathsf{Bool})$
- $(\mathsf{flatten}\ \mathsf{Bool} : \mathsf{Tree}\ \mathsf{Bool} \to \mathsf{List}\ \mathsf{Bool}) \frown_0$
  $(\mathsf{flatten}\ \mathbb{N} : \mathsf{Tree}\ \mathbb{N} \to \mathsf{List}\ \mathbb{N})$
  for any proof of $\mathsf{Bool} \frown_1 \mathbb{N}$

1-relatedness:

- $(\mathsf{List}_4\ \mathsf{Bool} : \mathcal{U}) \frown_1 (\mathsf{List}_7\ \mathsf{Bool} : \mathcal{U})$
- $(\mathsf{Bool} : \mathcal{U}) \frown_1 (\mathbb{N} : \mathcal{U})$ non-canonically
  (e.g. by setting $\mathsf{true} \frown_0 5$ and $\mathsf{false} \frown_0 2k+1$)

2-relatedness:

- $(\mathsf{Monoid} : \mathcal{U}) \frown_2 (\mathsf{Group} : \mathcal{U})$
  by setting $(M : \mathsf{Monoid}) \frown_1 (G : \mathsf{Group})$
  whenever $(M : \mathsf{Monoid}) \frown_1 (\mathsf{asMonoid}\ G : \mathsf{Monoid})$.

...

Examples

0-relatedness:

- $(2+5:\mathbb{N}) \frown_0 (7:\mathbb{N})$
- $([\mathsf{true},\mathsf{false}] : \mathsf{List}_4\ \mathsf{Bool}) \frown_0 ([\mathsf{true},\mathsf{false}] : \mathsf{List}_7\ \mathsf{Bool})$
- $(\mathsf{flatten}\ \mathsf{Bool} : \mathsf{Tree}\ \mathsf{Bool} \to \mathsf{List}\ \mathsf{Bool}) \frown_0$
  $(\mathsf{flatten}\ \mathbb{N} : \mathsf{Tree}\ \mathbb{N} \to \mathsf{List}\ \mathbb{N})$
  for any proof of $\mathsf{Bool} \frown_1 \mathbb{N}$

1-relatedness:

- $(\mathsf{List}_4\ \mathsf{Bool} : \mathcal{U}) \frown_1 (\mathsf{List}_7\ \mathsf{Bool} : \mathcal{U})$
- $(\mathsf{Bool} : \mathcal{U}) \frown_1 (\mathbb{N} : \mathcal{U})$ non-canonically
  (e.g. by setting $\mathsf{true} \frown_0 5$ and $\mathsf{false} \frown_0 2k+1$)

2-relatedness:

- $(\mathsf{Monoid} : \mathcal{U}) \frown_2 (\mathsf{Group} : \mathcal{U})$
  by setting $(M : \mathsf{Monoid}) \frown_1 (G : \mathsf{Group})$
  whenever $(M : \mathsf{Monoid}) \frown_1 (\mathsf{asMonoid}\ G : \mathsf{Monoid})$.

...

Depth

Depth of a type determines **amount of relations**:

- Depth -1: Unit, Empty, $P \vee Q$, $\ldots$
  $\top$
- Depth 0 (only equality): Bool, $\mathbb{N}$, List_n Bool, $\mathcal{U}^{-1}$, $\ldots$
  $a \frown_0 b \Rightarrow \top$
- Depth 1: $\mathcal{U}^0$, $\mathcal{U}^0 \to \mathcal{U}^0$, Group, Monoid, $\ldots$
  $a \frown_0 b \Rightarrow a \frown_1 b \Rightarrow \top$
- Depth 2: $\mathcal{U}^1$, $\ldots$
  $a \frown_0 b \Rightarrow a \frown_1 b \Rightarrow a \frown_2 b \Rightarrow \top$
- $\ldots$

Depth of a type determines **amount of relations**:

- Depth -1: Unit, Empty, $P \vee Q$, ...
  $\top$
- Depth 0 (only equality): Bool, $\mathbb{N}$, List_n Bool, $\mathcal{U}^{-1}$, ...
  $a \frown_0 b \Rightarrow \top$
- Depth 1: $\mathcal{U}^0$, $\mathcal{U}^0 \to \mathcal{U}^0$, Group, Monoid, ...
  $a \frown_0 b \Rightarrow a \frown_1 b \Rightarrow \top$
- Depth 2: $\mathcal{U}^1$, ...
  $a \frown_0 b \Rightarrow a \frown_1 b \Rightarrow a \frown_2 b \Rightarrow \top$
- ...

Depth of a type determines **amount of relations**:

- Depth -1: Unit, Empty, $P \vee Q$, ...
  $\top$
- Depth 0 (only equality): Bool, $\mathbb{N}$, List_n Bool, $\mathcal{U}^{-1}$, ...
  $a \frown_0 b \Rightarrow \top$
- Depth 1: $\mathcal{U}^0$, $\mathcal{U}^0 \to \mathcal{U}^0$, Group, Monoid, ...
  $a \frown_0 b \Rightarrow a \frown_1 b \Rightarrow \top$
- Depth 2: $\mathcal{U}^1$, ...
  $a \frown_0 b \Rightarrow a \frown_1 b \Rightarrow a \frown_2 b \Rightarrow \top$
- ...

Depth

Depth of a type determines **amount of relations**:

- Depth -1: Unit, Empty, $P \vee Q$, $\dots$
  $\top$
- Depth 0 (only equality): Bool, $\mathbb{N}$, List$_n$ Bool, $\mathcal{U}^{-1}$, $\dots$
  $a \frown_0 b \Rightarrow \top$
- Depth 1: $\mathcal{U}^0$, $\mathcal{U}^0 \to \mathcal{U}^0$, Group, Monoid, $\dots$
  $a \frown_0 b \Rightarrow a \frown_1 b \Rightarrow \top$
- Depth 2: $\mathcal{U}^1$, $\dots$
  $a \frown_0 b \Rightarrow a \frown_1 b \Rightarrow a \frown_2 b \Rightarrow \top$
- $\dots$

Depth

Depth of a type determines **amount of relations**:

- Depth -1: Unit, Empty, $P \vee Q$, ...
  $\top$
- Depth 0 (only equality): Bool, $\mathbb{N}$, List_n Bool, $\mathcal{U}^{-1}$, ...
  $a \frown_0 b \Rightarrow \top$
- Depth 1: $\mathcal{U}^0$, $\mathcal{U}^0 \to \mathcal{U}^0$, Group, Monoid, ...
  $a \frown_0 b \Rightarrow a \frown_1 b \Rightarrow \top$
- Depth 2: $\mathcal{U}^1$, ...
  $a \frown_0 b \Rightarrow a \frown_1 b \Rightarrow a \frown_2 b \Rightarrow \top$
- ...

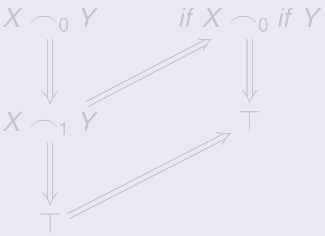
Depth of a type determines **amount of relations**:

- Depth -1: Unit, Empty, $P \vee Q$, $\ldots$
  $\top$
- Depth 0 (only equality): Bool, $\mathbb{N}$, List_n Bool, $\mathcal{U}^{-1}$, $\ldots$
  $a \frown_0 b \Rightarrow \top$
- Depth 1: $\mathcal{U}^0$, $\mathcal{U}^0 \to \mathcal{U}^0$, Group, Monoid, $\ldots$
  $a \frown_0 b \Rightarrow a \frown_1 b \Rightarrow \top$
- Depth 2: $\mathcal{U}^1$, $\ldots$
  $a \frown_0 b \Rightarrow a \frown_1 b \Rightarrow a \frown_2 b \Rightarrow \top$
- $\ldots$

Understanding modalities (1)

par : $1 \to 0$

$$\mathit{if} : (\mathbf{par} \mid X : \mathcal{U}^0) \to \mathrm{Bool} \to X \to X \to X$$



cnt : $1 \to 1$

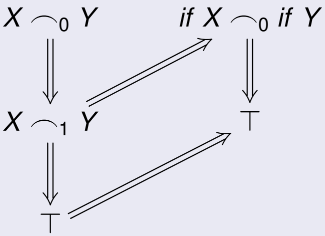
$$T : \mathcal{U}^0 \to \mathcal{U}^0$$

$$T\ X = \mathrm{Bool} \to X \to X \to X$$

Understanding modalities (1)

par : $1 \to 0$

$$if : (\mathbf{par} \mid X : \mathcal{U}^0) \to \mathrm{Bool} \to X \to X \to X$$



cnt : $1 \to 1$

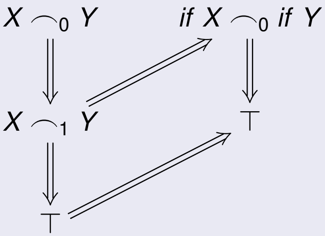
$$T : \mathcal{U}^0 \to \mathcal{U}^0$$

$$T\ X = \mathrm{Bool} \to X \to X \to X$$

Understanding modalities (1)

par : 1 → 0

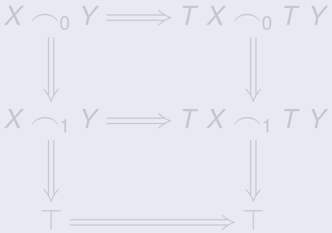
$$if : (\mathbf{par} \mid X : \mathcal{U}^0) \to \text{Bool} \to X \to X \to X$$



cnt : 1 → 1

$$T : \mathcal{U}^0 \to \mathcal{U}^0$$

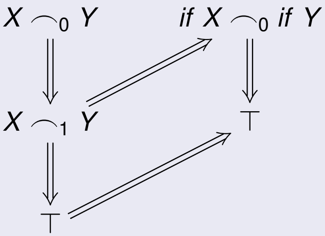
$$T\ X = \text{Bool} \to X \to X \to X$$



Understanding modalities (1)

par : 1 → 0

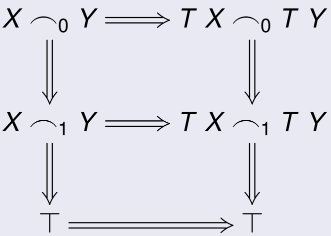
$$if : (\mathbf{par} \mid X : \mathcal{U}^0) \to \mathrm{Bool} \to X \to X \to X$$



cnt : 1 → 1

$$T : \mathcal{U}^0 \to \mathcal{U}^0$$

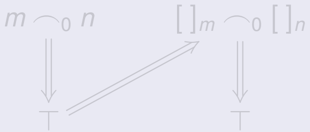
$$T\ X = \mathrm{Bool} \to X \to X \to X$$



Understanding modalities (2)

irr : $0 \to 0$

$$[\,] : (\mathbf{irr} \mid n : \mathbb{N}) \to \mathrm{List}_n\ A$$



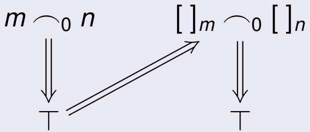
shi : $0 \to 1$

$$\lambda n.\mathrm{List}_n\ A : (\mathbf{shi} \mid n : \mathbb{N}) \to \mathcal{U}^0$$

Understanding modalities (2)

irr : $0 \to 0$

$$[\,] : (\mathbf{irr} \mid n : \mathbb{N}) \to \mathrm{List}_n\ A$$



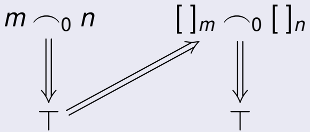
shi : $0 \to 1$

$$\lambda n.\mathrm{List}_n\ A : (\mathbf{shi} \mid n : \mathbb{N}) \to \mathcal{U}^0$$

Understanding modalities (2)

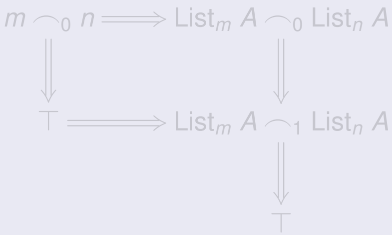
irr $: 0 \to 0$

$$[\,] : (\mathbf{irr} \mid n : \mathbb{N}) \to \mathrm{List}_n\ A$$



shi $: 0 \to 1$

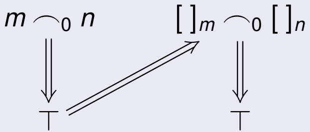
$$\lambda n.\mathrm{List}_n\ A : (\mathbf{shi} \mid n : \mathbb{N}) \to \mathcal{U}^0$$



Understanding modalities (2)

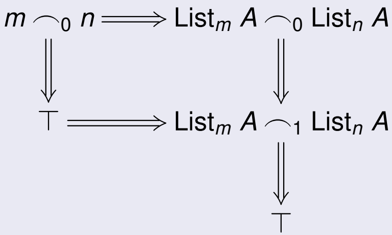
irr $: 0 \to 0$

$$[\,] : (\mathbf{irr} \mid n : \mathbb{N}) \to \mathrm{List}_n\ A$$



shi $: 0 \to 1$

$$\lambda n.\mathrm{List}_n\ A : (\mathbf{shi} \mid n : \mathbb{N}) \to \mathcal{U}^0$$



Composition of modalities

$\mathit{if}\ (\mathrm{List}_n\ A)\ b\ \mathit{as}\ [\ ]_n$

Irrelevant in n?

Yes if $\mathbf{par} \circ \mathbf{shi} = \mathbf{irr} : 0 \to 0$

if (List$_n$ A) b *as* $[\,]_n$

Irrelevant in n?

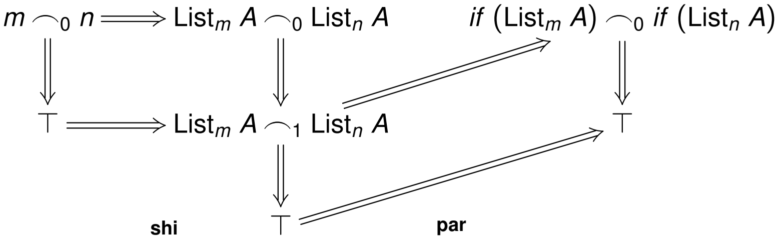
Yes if $\textbf{par} \circ \textbf{shi} = \textbf{irr} : 0 \to 0$

Composition of modalities

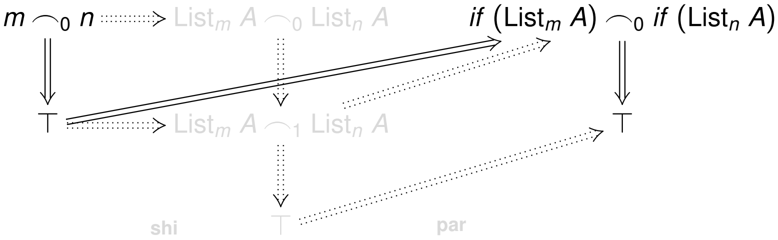
$\mathit{if}\ (\mathrm{List}_n\ A)\ b\ \mathit{as}\ [\ ]_n$

Irrelevant in n?

Yes if $\mathbf{par} \circ \mathbf{shi} = \mathbf{irr} : 0 \to 0$



if (List_n A) *b as* $[\,]_n$
Irrelevant in n?
Yes if $\mathbf{par} \circ \mathbf{shi} = \mathbf{irr} : 0 \to 0$



Conclusion

- Unified framework (type system + presheaf model) for:
  - **parametricity**
  - **ad hoc polymorphism**
  - **. irrelevance**
  - **.. shape-irrelevance**
  - aspects of **unions, intersections, algebra, Prop, . . .**

  Continuing Nuyts, Vezzosi, Devriese (2017)
- Understanding of **modality** of **function** $\leftrightarrow$ dependent **codomain**
- **Depth** explains which modalities apply given the types

  See Licata et al. (2016, 2017) for multi-mode type theory
- Type-checking time **erasure of irrelevant subterms**

Conclusion

- Unified framework (type system + presheaf model) for:
  - **parametricity**
  - **ad hoc polymorphism**
  - **. irrelevance**
  - **.. shape-irrelevance**
  - aspects of **unions, intersections, algebra, Prop, . . .**

  Continuing Nuyts, Vezzosi, Devriese (2017)
- Understanding of **modality** of **function** $\leftrightarrow$ dependent **codomain**
- **Depth** explains which modalities apply given the types

  See Licata et al. (2016, 2017) for multi-mode type theory
- Type-checking time **erasure of irrelevant subterms**

Conclusion

- Unified framework (type system + presheaf model) for:
  - **parametricity**
  - **ad hoc polymorphism**
  - **. irrelevance**
  - **.. shape-irrelevance**
  - aspects of **unions, intersections, algebra, Prop, . . .**

  Continuing Nuyts, Vezzosi, Devriese (2017)
- Understanding of **modality** of **function** $\leftrightarrow$ dependent **codomain**
- **Depth** explains which modalities apply given the types

  See Licata et al. (2016, 2017) for multi-mode type theory
- Type-checking time **erasure of irrelevant subterms**

Conclusion

- Unified framework (type system + presheaf model) for:
  - **parametricity**
  - **ad hoc polymorphism**
  - **. irrelevance**
  - **.. shape-irrelevance**
  - aspects of **unions, intersections, algebra, Prop, . . .**

  Continuing Nuyts, Vezzosi, Devriese (2017)
- Understanding of **modality** of **function** $\leftrightarrow$ dependent **codomain**
- **Depth** explains which modalities apply given the types

  See Licata et al. (2016, 2017) for multi-mode type theory
- Type-checking time **erasure of irrelevant subterms**

Thanks!

Questions?