Transpension: The Right Adjoint to the Π-type

Andreas Nuyts, Dominique Devriese

KU Leuven, Belgium

TYPES '22
Nantes, France
June 21, 2022

Presheaf semantics can model:

- **HoTT** (preservation of **isomorphisms**),
- **Parametricity** (preservation of **relations**),
- **Guarded TT** (preservation of **stage of computation**),
- **Nominal TT** (preservation of **renaming** and α**-equivalence**),
- **Directed TT** (preservation of **homomorphisms**).

Use these preservation properties **within type theory**?

Presheaf semantics can model:

- **HoTT** (preservation of **isomorphisms**),
- **Parametricity** (preservation of **relations**),
- **Guarded TT** (preservation of **stage of computation**),
- **Nominal TT** (preservation of **renaming** and α-**equivalence**),
- **Directed TT** (preservation of **homomorphisms**).

Use these preservation properties **within type theory**?

Presheaf semantics can model:

- **HoTT** (preservation of **isomorphisms**),
- **Parametricity** (preservation of **relations**),
- **Guarded TT** (preservation of **stage of computation**),
- **Nominal TT** (preservation of **renaming** and α**-equivalence**),
- **Directed TT** (preservation of **homomorphisms**).

Use these preservation properties **within type theory**?

Presheaf semantics can model:

- **HoTT** (preservation of **isomorphisms**),
- **Parametricity** (preservation of **relations**),
- **Guarded TT** (preservation of **stage of computation**),
- **Nominal TT** (preservation of **renaming** and α**-equivalence**),
- **Directed TT** (preservation of **homomorphisms**).

Use these preservation properties **within type theory**?

Presheaf semantics can model:

- **HoTT** (preservation of **isomorphisms**),
- **Parametricity** (preservation of **relations**),
- **Guarded TT** (preservation of **stage of computation**),
- **Nominal TT** (preservation of **renaming** and α**-equivalence**),
- **Directed TT** (preservation of **homomorphisms**).

Use these preservation properties **within type theory**?

Presheaf semantics can model:

- **HoTT** (preservation of **isomorphisms**),
- **Parametricity** (preservation of **relations**),
- **Guarded TT** (preservation of **stage of computation**),
- **Nominal TT** (preservation of **renaming** and α**-equivalence**),
- **Directed TT** (preservation of **homomorphisms**).

Use these preservation properties **within type theory**?

Internalization Operators

We want **simpler foundations:**

- simpler metatheory,
- cross-fertilization (e.g. non-affine Ψ/Gel),
- guidance (e.g. directed TT).

Transpension		Strictness	Pushout
$\forall u \dashv \between u$		(OP18)	
(Yet87, ND22)			

HoTT $(\mathbb{I} \to \sqcup) \dashv \surd$ (LOPS18)			HoTT/DirTT/Param. Glue (CCHM16/ NVD17/WL20)	Param. Weld (NVD17)
Nominal TT $\reflectbox{N} i, \langle\langle i \rangle\rangle, \nu\, i$ (PMD14)	Func. param. Φ/extent (BCM15)	Type param. Ψ/Gel (BCM15)	mill Weld $\leftrightarrow \forall i$ (ND18)	Guarded TT Löb ind.

Internalization Operators

We want **simpler foundations:**

- simpler metatheory,
- cross-fertilization (e.g. non-affine Ψ/Gel),
- guidance (e.g. directed TT).

Transpension
$\forall u \dashv ≬ u$
(Yet87, ND22)

Strictness
(OP18)

Pushout

HoTT
$(\mathbb{I} \to \sqcup) \dashv \surd$
(LOPS18)

HoTT/DirTT/Param.
Glue
(CCHM16/
NVD17/WL20)

Param.
Weld
(NVD17)

Nominal TT
$Иi, \langle\langle i \rangle\rangle, \nu i$
(PMD14)

Func. param.
Φ/extent
(BCM15)

Type param.
Ψ/Gel
(BCM15)

mill
Weld $\leftrightarrow \forall i$
(ND18)

Guarded TT
Löb ind.

Internalization Operators

We want **simpler foundations:**

- simpler metatheory,
- cross-fertilization (e.g. non-affine Ψ/Gel),
- guidance (e.g. directed TT).

Transpension
$\forall u \dashv \between u$
(Yet87, ND22)

Strictness
(OP18)

↓

HoTT/DirTT/Param.
Glue
(CCHM16/
NVD17/WL20)

Pushout

Param.
Weld
(NVD17)

HoTT
$(\mathbb{I} \to \sqcup) \dashv \surd$
(LOPS18)

Nominal TT
$Иi, \langle\langle i \rangle\rangle, \nu i$
(PMD14)

Func. param.
Φ/extent
(BCM15)

Type param.
Ψ/Gel
(BCM15)

mill
Weld $\leftrightarrow \forall i$
(ND18)

Guarded TT
Löb ind.

Internalization Operators

We want **simpler foundations:**

- simpler metatheory,
- cross-fertilization (e.g. non-affine Ψ/Gel),
- guidance (e.g. directed TT).

Transpension
$\forall u \dashv \between u$
(Yet87, ND22)

Strictness
(OP18)

Pushout

HoTT
$(\mathbb{I} \to \sqcup) \dashv \surd$
(LOPS18)

HoTT/DirTT/Param.
Glue
(CCHM16/
NVD17/WL20)

Param.
Weld
(NVD17)

Nominal TT
$\text{И}\, i, \langle\langle i \rangle\rangle, \nu\, i$
(PMD14)

Func. param.
Φ/extent
(BCM15)

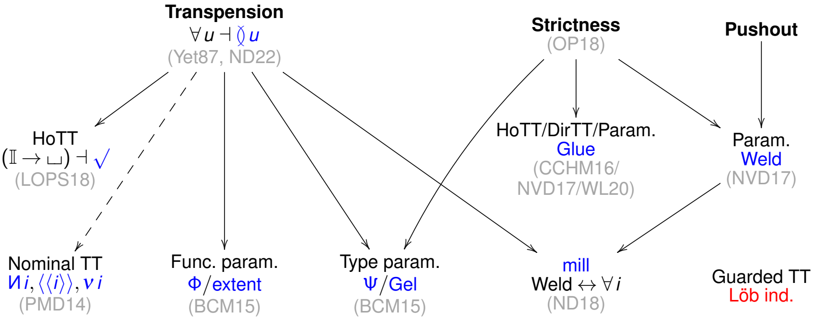
Type param.
Ψ/Gel
(BCM15)

mill
Weld $\leftrightarrow \forall i$
(ND18)

Guarded TT
Löb ind.

We want **simpler foundations:**

- simpler metatheory,
- cross-fertilization (e.g. non-affine Ψ/Gel),
- guidance (e.g. directed TT).



LMCS submission: `https://arxiv.org/abs/2008.08533`

- Type system with transpension $\between u$ for shape variable $u : \mathbb{U}$, e.g.
  - $i : \mathbb{I}$ in HoTT/Param.,
  - $i : \mathbf{2}$ in Directed TT,
  - $i : N$ in Nominal TT,
- Built on an instance of **MTT** (multimodal TT),
- With presheaf semantics,
- No type-checking algorithm.

Our contribution

LMCS submission: `https://arxiv.org/abs/2008.08533`

- Type system with transpension $\between u$ for shape variable $u : \mathbb{U}$, e.g.
  - $i : \mathbb{I}$ in HoTT/Param.,
  - $i : \mathbf{2}$ in Directed TT,
  - $i : N$ in Nominal TT,
- Built on an instance of **MTT** (multimodal TT),
- With presheaf semantics,
- No type-checking algorithm.

Our contribution

LMCS submission: https://arxiv.org/abs/2008.08533

- Type system with transpension $\between u$ for shape variable $u : \mathbb{U}$, e.g.
  - $i : \mathbb{I}$ in HoTT/Param.,
  - $i : \mathbf{2}$ in Directed TT,
  - $i : N$ in Nominal TT,
- Built on an instance of **MTT** (multimodal TT),
- With presheaf semantics,
- No type-checking algorithm.

Our contribution

LMCS submission: https://arxiv.org/abs/2008.08533

- Type system with transpension $≬u$ for shape variable $u : \mathbb{U}$, e.g.
  - $i : \mathbb{I}$ in HoTT/Param.,
  - $i : \mathbf{2}$ in Directed TT,
  - $i : N$ in Nominal TT,
- Built on an instance of **MTT** (multimodal TT),
- With presheaf semantics,
- No type-checking algorithm.

Our contribution

LMCS submission: https://arxiv.org/abs/2008.08533

- Type system with transpension $\between u$ for shape variable $u : \mathbb{U}$, e.g.
  - $i : \mathbb{I}$ in HoTT/Param.,
  - $i : \mathbf{2}$ in Directed TT,
  - $i : N$ in Nominal TT,
- Built on an instance of **MTT** (multimodal TT),
- With presheaf semantics,
- No type-checking algorithm.

MTT is parametrized by a **2-category**:

- modes $p, q, r, \ldots$
- modalities $\mu : p \to q$,
  - On types $T \mapsto \langle \mu \mid T \rangle$,
  - On contexts $\Gamma \mapsto (\Gamma, 🔒_\mu)$,
  - $🔒_\mu \dashv \mu$.
- (2-cells $\alpha : \mu \Rightarrow \nu$).

Typical **semantics:**

- $[\![p]\!]$ is a presheaf category modelling all of DTT,
- $[\![🔒_\mu]\!] \dashv [\![\mu]\!]$ is a dependent ~~right~~ adjunction (DRA) (BCMMPS20),

Multimodal TT (MTT)

MTT is parametrized by a **2-category**:

- modes p, q, r, ...
- modalities $\mu : p \to q$,
  - On types $T \mapsto \langle \mu \mid T \rangle$,
  - On contexts $\Gamma \mapsto (\Gamma, 🔒_\mu)$,
  - $🔒_\mu \dashv \mu$.
- (2-cells $\alpha : \mu \Rightarrow \nu$).

Typical **semantics:**

- $[\![p]\!]$ is a presheaf category modelling all of DTT,
- $[\![🔒_\mu]\!] \dashv [\![\mu]\!]$ is a dependent ~~right~~ adjunction (DRA) (BCMMPS20),

Multimodal TT (MTT)

MTT is parametrized by a **2-category**:

- modes $p, q, r, \ldots$
- modalities $\mu : p \to q$,
  - On types $T \mapsto \langle \mu \mid T \rangle$,
  - On contexts $\Gamma \mapsto (\Gamma, \text{🔒}_\mu)$,
  - $\text{🔒}_\mu \dashv \mu$.
- (2-cells $\alpha : \mu \Rightarrow \nu$).

Typical **semantics:**

- $[\![p]\!]$ is a presheaf category modelling all of DTT,
- $[\![\text{🔒}_\mu]\!] \dashv [\![\mu]\!]$ is a dependent ~~right~~ adjunction (DRA) (BCMMPS20),

MTT is parametrized by a **2-category**:

- modes $p, q, r, \ldots$
- modalities $\mu : p \to q$,
  - On types $T \mapsto \langle \mu \mid T \rangle$,
  - On contexts $\Gamma \mapsto (\Gamma, 🔒_\mu)$,
  - $🔒_\mu \dashv \mu$.
- (2-cells $\alpha : \mu \Rightarrow \nu$).

Typical **semantics:**

- $[\![p]\!]$ is a presheaf category modelling all of DTT,
- $[\![🔒_\mu]\!] \dashv [\![\mu]\!]$ is a dependent ~~right~~ adjunction (DRA) (BCMMPS20),

Multimodal TT (MTT)

MTT is parametrized by a **2-category**:

- modes $p, q, r, \ldots$
- modalities $\mu : p \to q$,
  - On types $T \mapsto \langle \mu \mid T \rangle$,
  - On contexts $\Gamma \mapsto (\Gamma, 🔒_{\mu})$,
  - $🔒_{\mu} \dashv \mu$.
- (2-cells $\alpha : \mu \Rightarrow \nu$).

Typical **semantics:**

- $[\![p]\!]$ is a presheaf category modelling all of DTT,
- $[\![🔒_{\mu}]\!] \dashv [\![\mu]\!]$ is a dependent ~~right~~ adjunction (DRA) (BCMMPS20),

MTT Instantiation & Presheaf Semantics

Pick a base category $\mathscr{W}$.
$\Rightarrow$ We get a presheaf model $\mathrm{Psh}(\mathscr{W})$.

MTT Instantiation & Presheaf Semantics

Pick a base category $\mathscr{W}$.
$\Rightarrow$ We get a presheaf model $\mathrm{Psh}(\mathscr{W})$.

- **Problem:** Modalities $\forall(u : \mathbb{U}) \dashv \between u$ bind / depend on u (unsupported by MTT).
- **Solution: Modes** are shape contexts
  e.g. $\Xi = (u : \mathbb{U}, v : \mathbb{V}, w : \mathbb{W})$
  Formally, Ξ is a context in $\mathrm{Psh}(\mathscr{W})$.
- **Judgements at mode** Ξ are modelled in $\mathrm{Psh}(\int_{\mathscr{W}} \Xi)$:

TT in $\mathrm{Psh}(\int_{\mathscr{W}} \Xi)$		TT in $\mathrm{Psh}(\mathscr{W})$
		Ξ ctx
Γ ctx	$\sim$	$\Xi.\Gamma$ ctx
$\Gamma \vdash T$ type	$\sim$	$\Xi.\Gamma \vdash T$ type
$\Gamma \vdash t : T$	$\sim$	$\Xi.\Gamma \vdash t : T$

$\Rightarrow$ We're doing type theory in a fixed context Ξ.

Modes

- **Problem:** Modalities $\forall(u : \mathbb{U}) \dashv \between u$ bind / depend on u (unsupported by MTT).
- **Solution:** **Modes** are shape contexts
  e.g. $\Xi = (u : \mathbb{U}, v : \mathbb{V}, w : \mathbb{W})$
  Formally, Ξ is a context in $\mathrm{Psh}(\mathscr{W})$.
- **Judgements at mode** Ξ are modelled in $\mathrm{Psh}(\int_{\mathscr{W}} \Xi)$:

TT in $\mathrm{Psh}(\int_{\mathscr{W}} \Xi)$		TT in $\mathrm{Psh}(\mathscr{W})$
		Ξ ctx
Γ ctx	$\sim$	$\Xi.\Gamma$ ctx
$\Gamma \vdash T$ type	$\sim$	$\Xi.\Gamma \vdash T$ type
$\Gamma \vdash t : T$	$\sim$	$\Xi.\Gamma \vdash t : T$

$\Rightarrow$ We're doing type theory in a fixed context Ξ.

Modes

- **Problem:** Modalities $\forall(u : \mathbb{U}) \dashv \between u$ bind / depend on u (unsupported by MTT).
- **Solution: Modes** are shape contexts
  e.g. $\Xi = (u : \mathbb{U}, v : \mathbb{V}, w : \mathbb{W})$
  Formally, Ξ is a context in $\mathrm{Psh}(\mathscr{W})$.
- **Judgements at mode** Ξ are modelled in $\mathrm{Psh}(\int_{\mathscr{W}} \Xi)$:

TT in $\mathrm{Psh}(\int_{\mathscr{W}} \Xi)$		TT in $\mathrm{Psh}(\mathscr{W})$
		Ξ ctx
Γ ctx	$\sim$	$\Xi.\Gamma$ ctx
$\Gamma \vdash T$ type	$\sim$	$\Xi.\Gamma \vdash T$ type
$\Gamma \vdash t : T$	$\sim$	$\Xi.\Gamma \vdash t : T$

$\Rightarrow$ We're doing type theory in a fixed context Ξ.

- **Problem:** Modalities $\forall(u : \mathbb{U}) \dashv \between u$ bind / depend on u (unsupported by MTT).
- **Solution: Modes** are shape contexts
  e.g. $\Xi = (u : \mathbb{U}, v : \mathbb{V}, w : \mathbb{W})$
  Formally, Ξ is a context in $\mathrm{Psh}(\mathscr{W})$.
- **Judgements at mode** Ξ are modelled in $\mathrm{Psh}(\int_{\mathscr{W}} \Xi)$:

$$\frac{\text{TT in } \mathrm{Psh}(\int_{\mathscr{W}} \Xi) \qquad \text{TT in } \mathrm{Psh}(\mathscr{W})}{}$$

TT in $\mathrm{Psh}(\int_{\mathscr{W}} \Xi)$		TT in $\mathrm{Psh}(\mathscr{W})$
		Ξ ctx
Γ ctx	$\sim$	$\Xi.\Gamma$ ctx
$\Gamma \vdash T$ type	$\sim$	$\Xi.\Gamma \vdash T$ type
$\Gamma \vdash t : T$	$\sim$	$\Xi.\Gamma \vdash t : T$

$\Rightarrow$ We're doing type theory in a fixed context Ξ.

Shapes/Multipliers

- A **(substructural?) shape** is an $U \in \mathscr{W}$ together with an **arbitrary** "multiplier" functor $\sqcup \ltimes U : \mathscr{W} \to \mathscr{W}$ such that $U \cong \top \ltimes U$.

  E.g. twisted prism functor (PK19)

- A **cartesian shape** is a $U \in \mathscr{W}$ where $\mathscr{W}$ is cartesian.
  $\to$ We get "multiplier" $\sqcup \times U : \mathscr{W} \to \mathscr{W}$.

Multiplier extends over $\mathbf{y} : \mathscr{W} \subseteq \mathrm{Psh}(\mathscr{W})$ as $\sqcup \ltimes \mathbb{U} : \mathrm{Psh}(\mathscr{W}) \to \mathrm{Psh}(\mathscr{W}) : \Xi \mapsto (\Xi, u : \mathbb{U})$ (**shape context extension**).

Assume multiplier is **local right adjoint**:

$$\exists_U : \mathscr{W} \to \mathscr{W}/U : W \mapsto (W \ltimes U, \pi_2)$$

should have left adjoint $\exists_U \dashv \exists_U$.

Shapes/Multipliers

- A **(substructural?) shape** is an $U \in \mathscr{W}$ together with an **arbitrary** "multiplier" functor $\sqcup \ltimes U : \mathscr{W} \to \mathscr{W}$ such that $U \cong \top \ltimes U$.

  E.g. twisted prism functor (PK19)

- A **cartesian shape** is a $U \in \mathscr{W}$ where $\mathscr{W}$ is cartesian.
  $\to$ We get "multiplier" $\sqcup \times U : \mathscr{W} \to \mathscr{W}$.

Multiplier extends over $\mathbf{y} : \mathscr{W} \subseteq \mathrm{Psh}(\mathscr{W})$ as
$\sqcup \ltimes \mathbb{U} : \mathrm{Psh}(\mathscr{W}) \to \mathrm{Psh}(\mathscr{W}) : \Xi \mapsto (\Xi, u : \mathbb{U})$
(**shape context extension**).

Assume multiplier is **local right adjoint**:

$$\exists_U : \mathscr{W} \to \mathscr{W}/U : W \mapsto (W \ltimes U, \pi_2)$$

should have left adjoint $\exists_U \dashv \exists_U$.

Shapes/Multipliers

- A **(substructural?) shape** is an $U \in \mathscr{W}$ together with an **arbitrary** "multiplier" functor $\sqcup \ltimes U : \mathscr{W} \to \mathscr{W}$ such that $U \cong \top \ltimes U$.

  E.g. twisted prism functor (PK19)

- A **cartesian shape** is a $U \in \mathscr{W}$ where $\mathscr{W}$ is cartesian.
  $\to$ We get "multiplier" $\sqcup \times U : \mathscr{W} \to \mathscr{W}$.

Multiplier extends over $\mathbf{y} : \mathscr{W} \subseteq \mathrm{Psh}(\mathscr{W})$ as
$\sqcup \ltimes \mathbb{U} : \mathrm{Psh}(\mathscr{W}) \to \mathrm{Psh}(\mathscr{W}) : \Xi \mapsto (\Xi, u : \mathbb{U})$
(**shape context extension**).

Assume multiplier is **local right adjoint**:

$$\dashv_U : \mathscr{W} \to \mathscr{W}/U : W \mapsto (W \ltimes U, \pi_2)$$

should have left adjoint $\exists_U \dashv \dashv_U$.

Shapes/Multipliers

- A **(substructural?) shape** is an $U \in \mathscr{W}$ together with an **arbitrary** "multiplier" functor $\sqcup \ltimes U : \mathscr{W} \to \mathscr{W}$ such that $U \cong \top \ltimes U$.

  E.g. twisted prism functor (PK19)
- A **cartesian shape** is a $U \in \mathscr{W}$ where $\mathscr{W}$ is cartesian.
  $\to$ We get "multiplier" $\sqcup \times U : \mathscr{W} \to \mathscr{W}$.

Multiplier extends over $\mathbf{y} : \mathscr{W} \subseteq \mathrm{Psh}(\mathscr{W})$ as
$\sqcup \ltimes \mathbb{U} : \mathrm{Psh}(\mathscr{W}) \to \mathrm{Psh}(\mathscr{W}) : \Xi \mapsto (\Xi, u : \mathbb{U})$
(**shape context extension**).

Assume multiplier is **local right adjoint**:

$$\exists_U : \mathscr{W} \to \mathscr{W}/U : W \mapsto (W \ltimes U, \pi_2)$$

should have left adjoint $\exists_U \dashv \exists_U$.

We get

$$\exists^{\int \Xi}_{\mathbb{U}} \dashv \reflectbox{$\exists$}^{\int \Xi}_{\mathbb{U}} : \int_{\mathscr{W}} \Xi \to \int_{\mathscr{W}} (\Xi, u : \mathbb{U}),$$

whence 4 presheaf functors and 3 modalities

$$\begin{array}{ccccccc}
(\exists^{\int \Xi}_{\mathbb{U}})_! & \dashv & (\exists^{\int \Xi}_{\mathbb{U}})^* & \dashv & (\exists^{\int \Xi}_{\mathbb{U}})_* & & \\
 & & \| \wr & & \| \wr & & \\
 & & (\reflectbox{$\exists$}^{\int \Xi}_{\mathbb{U}})_! & \dashv & (\reflectbox{$\exists$}^{\int \Xi}_{\mathbb{U}})^* & \dashv & (\reflectbox{$\exists$}^{\int \Xi}_{\mathbb{U}})_* \\
\hline\hline
\exists^{\Xi}_{\mathbb{U}} & \dashv & \reflectbox{$\exists$}^{\Xi}_{\mathbb{U}} & \dashv & \forall^{\Xi}_{\mathbb{U}} & \dashv & \between^{\Xi}_{\mathbb{U}} \\
\hline\hline
[\![\text{🔒}_{\reflectbox{$\exists$} u}]\!] & \dashv & [\![\text{🔒}_{\forall u}]\!] & \dashv & [\![\text{🔒}_{\between u}]\!] & & \\
 & & [\![\reflectbox{$\exists$} u]\!] & \dashv & [\![\forall u]\!] & \dashv & [\![\between u]\!]
\end{array}$$

We get

$$\exists_{\mathbb{U}}^{\int \Xi} \dashv Ⅎ_{\mathbb{U}}^{\int \Xi} : \int_{\mathscr{W}} \Xi \to \int_{\mathscr{W}} (\Xi, u : \mathbb{U}),$$

whence 4 presheaf functors and 3 modalities

$$\begin{array}{ccccccccc}
(\exists_{\mathbb{U}}^{\int \Xi})_! & \dashv & (\exists_{\mathbb{U}}^{\int \Xi})^* & \dashv & (\exists_{\mathbb{U}}^{\int \Xi})_* & & \\
 & & \| \wr & & \| \wr & & \\
 & & (Ⅎ_{\mathbb{U}}^{\int \Xi})_! & \dashv & (Ⅎ_{\mathbb{U}}^{\int \Xi})^* & \dashv & (Ⅎ_{\mathbb{U}}^{\int \Xi})_* \\
\hline\hline
\exists_{\mathbb{U}}^{\Xi} & \dashv & Ⅎ_{\mathbb{U}}^{\Xi} & \dashv & \forall_{\mathbb{U}}^{\Xi} & \dashv & ≬_{\mathbb{U}}^{\Xi} \\
\hline\hline
[\![🔒_{Ⅎ u}]\!] & \dashv & [\![🔒_{\forall u}]\!] & \dashv & [\![🔒_{≬ u}]\!] & & \\
 & & [\![Ⅎ\, u]\!] & \dashv & [\![\forall\, u]\!] & \dashv & [\![≬\, u]\!]
\end{array}$$

Shape Weakening Modalities

From $\pi_1 : (\Xi, u : \mathbb{U}) \to \Xi$ whence

$$\Sigma_{\mathbb{U}}^{\int \Xi} : \int_{\mathscr{W}} (\Xi, u : \mathbb{U}) \to \int_{\mathscr{W}} \Xi,$$

whence 3 presheaf functors and 2 modalities

$$(\Sigma_{\mathbb{U}}^{\int \Xi})_! \dashv (\Sigma_{\mathbb{U}}^{\int \Xi})^* \dashv (\Sigma_{\mathbb{U}}^{\int \Xi})_*$$

$$\Sigma_{\mathbb{U}}^{\Xi} \dashv \Omega_{\mathbb{U}}^{\Xi} \dashv \Pi_{\mathbb{U}}^{\Xi}$$

$$[\![🔒_{\Omega\, u}]\!] \dashv [\![🔒_{\Pi\, u}]\!]$$

$$[\![\Omega\, u]\!] \dashv [\![\Pi\, u]\!]$$

Shape weakening $\Omega\, u$ is a modality (does not compute).
For **cartesian** multipliers, we get $Ⅎ\, u = \Omega\, u$ and $\forall\, u = \Pi\, u \dashv \between\, u$.

Shape Weakening Modalities

From $\pi_1 : (\Xi, u : \mathbb{U}) \to \Xi$ whence

$$\Sigma_{\mathbb{U}}^{\int \Xi} : \int_{\mathscr{W}} (\Xi, u : \mathbb{U}) \to \int_{\mathscr{W}} \Xi,$$

whence 3 presheaf functors and 2 modalities

$$(\Sigma_{\mathbb{U}}^{\int \Xi})_! \quad \dashv \quad (\Sigma_{\mathbb{U}}^{\int \Xi})^* \quad \dashv \quad (\Sigma_{\mathbb{U}}^{\int \Xi})_*$$

$$\Sigma_{\mathbb{U}}^{\Xi} \quad \dashv \quad \Omega_{\mathbb{U}}^{\Xi} \quad \dashv \quad \Pi_{\mathbb{U}}^{\Xi}$$

$$[\![\text{🔒}_{\Omega\, u}]\!] \quad \dashv \quad [\![\text{🔒}_{\sqcap\, u}]\!]$$

$$[\![\Omega\, u]\!] \quad \dashv \quad [\![\sqcap\, u]\!]$$

Shape weakening $\Omega\, u$ is a modality (does not compute).

For **cartesian** multipliers, we get $\exists\, u = \Omega\, u$ and $\forall\, u = \sqcap\, u \dashv \between\, u$.

Shape Weakening Modalities

From $\pi_1 : (\Xi, u : \mathbb{U}) \to \Xi$ whence

$$\Sigma_{\mathbb{U}}^{\int \Xi} : \int_{\mathscr{W}} (\Xi, u : \mathbb{U}) \to \int_{\mathscr{W}} \Xi,$$

whence 3 presheaf functors and 2 modalities

$$(\Sigma_{\mathbb{U}}^{\int \Xi})_! \dashv (\Sigma_{\mathbb{U}}^{\int \Xi})^* \dashv (\Sigma_{\mathbb{U}}^{\int \Xi})_*$$

$$\Sigma_{\mathbb{U}}^{\Xi} \dashv \Omega_{\mathbb{U}}^{\Xi} \dashv \Pi_{\mathbb{U}}^{\Xi}$$

$$[\![🔒_{\Omega\, u}]\!] \dashv [\![🔒_{\Pi\, u}]\!]$$

$$[\![\Omega\, u]\!] \dashv [\![\Pi\, u]\!]$$

Shape weakening $\Omega\, u$ is a modality (does not compute).

For **cartesian** multipliers, we get $\exists\, u = \Omega\, u$ and $\forall\, u = \Pi\, u \dashv \between u$.

Shape Weakening Modalities

From $\pi_1 : (\Xi, u : \mathbb{U}) \to \Xi$ whence

$$\Sigma_{\mathbb{U}}^{\int \Xi} : \int_{\mathscr{W}} (\Xi, u : \mathbb{U}) \to \int_{\mathscr{W}} \Xi,$$

whence 3 presheaf functors and 2 modalities

$$(\Sigma_{\mathbb{U}}^{\int \Xi})_! \quad \dashv \quad (\Sigma_{\mathbb{U}}^{\int \Xi})^* \quad \dashv \quad (\Sigma_{\mathbb{U}}^{\int \Xi})_*$$

$$\Sigma_{\mathbb{U}}^{\Xi} \quad \dashv \quad \Omega_{\mathbb{U}}^{\Xi} \quad \dashv \quad \Pi_{\mathbb{U}}^{\Xi}$$

$$[\![🔒_{\Omega\, u}]\!] \quad \dashv \quad [\![🔒_{\Pi\, u}]\!]$$

$$[\![\Omega\, u]\!] \quad \dashv \quad [\![\Pi\, u]\!]$$

Shape weakening $\Omega\, u$ is a modality (does not compute).

For **cartesian** multipliers, we get $\exists\, u = \Omega\, u$ and $\forall\, u = \Pi\, u \dashv \between\, u$.

Shape Substitution Modalities

From $\sigma : \Xi_1 \to \Xi_2$ (e.g. weakening), we get

$$\Sigma^{\int \sigma} : \int_{\mathscr{W}} \Xi_1 \to \int_{\mathscr{W}} \Xi_2,$$

whence 3 presheaf functors and 2 modalities

$$(\Sigma^{\int \sigma})_! \quad \dashv \quad (\Sigma^{\int \sigma})^* \quad \dashv \quad (\Sigma^{\int \sigma})_*$$

$$\Sigma^{\sigma} \quad \dashv \quad \Omega^{\sigma} \quad \dashv \quad \Pi^{\sigma}$$

$$[\![🔒_{\Omega\,\sigma}]\!] \quad \dashv \quad [\![🔒_{\Pi\,\sigma}]\!]$$

$$[\![\Omega\,\sigma]\!] \quad \dashv \quad [\![\Pi\,\sigma]\!]$$

Shape subst $\Omega\,\sigma$ is a modality (does not compute).

Conclusion

We can recover most internal presheaf operators using:

- The **transpension type**,
- The **strictness axiom**,
- A **pushout type**.

Basic semantics of transpension are straightforward.

Take home message

Check the preprint! `https://arxiv.org/abs/2008.08533`

Thanks!

Questions?

Conclusion

We can recover most internal presheaf operators using:

- The **transpension type**,
- The **strictness axiom**,
- A **pushout type**.

Basic semantics of transpension are straightforward.

Take home message

Check the preprint! `https://arxiv.org/abs/2008.08533`

Thanks!

Questions?

Quantification Theorem

- For **semicartesian** multipliers (where $\Omega\, u$ exists), we get

$$\mathsf{spoil}_u : \text{⅃}\, u \Rightarrow \Omega\, u \qquad \mathsf{cospoil}_u : \sqcap\, u \Rightarrow \forall\, u$$

- For **cartesian** multipliers, we get

$$\text{⅃}\, u = \Omega\, u \qquad \sqcap\, u = \forall\, u$$

- For **cancellative and affine** multipliers (where ⅃_U is **fully faithful**), we get invertible (co-)units

$$\mathsf{const}_u : 1 \cong \forall\, u \circ \text{⅃}\, u$$

$$\mathsf{unmer}_u : \forall\, u \circ \between\, u \cong 1$$

- For **cancellative, affine and connection-free** multipliers, we get Φ/extent.

Quantification Theorem

- For **semicartesian** multipliers (where $\Omega\, u$ exists), we get

$$\mathsf{spoil}_u : \exists\, u \Rightarrow \Omega\, u \qquad \mathsf{cospoil}_u : \Pi\, u \Rightarrow \forall\, u$$

- For **cartesian** multipliers, we get

$$\exists\, u = \Omega\, u \qquad \Pi\, u = \forall\, u$$

- For **cancellative and affine** multipliers (where $\exists_U$ is **fully faithful**), we get invertible (co-)units

$$\mathsf{const}_u : 1 \cong \forall\, u \circ \exists\, u$$

$$\mathsf{unmer}_u : \forall\, u \circ \between\, u \cong 1$$

- For **cancellative, affine and connection-free** multipliers, we get Φ / extent.

Quantification Theorem

- For **semicartesian** multipliers (where $\Omega\, u$ exists), we get

$$\text{spoil}_u : \Finv\, u \Rightarrow \Omega\, u \qquad \text{cospoil}_u : \Pi\, u \Rightarrow \forall\, u$$

- For **cartesian** multipliers, we get

$$\Finv\, u = \Omega\, u \qquad \Pi\, u = \forall\, u$$

- For **cancellative and affine** multipliers (where $\Finv_U$ is **fully faithful**), we get invertible (co-)units

$$\text{const}_u : 1 \cong \forall\, u \circ \Finv\, u$$

$$\text{unmer}_u : \forall\, u \circ \between\, u \cong 1$$

- For **cancellative, affine and connection-free** multipliers, we get Φ/extent.

Quantification Theorem

- For **semicartesian** multipliers (where $\Omega\, u$ exists), we get

$$\mathsf{spoil}_u : \Finv\, u \Rightarrow \Omega\, u \qquad \mathsf{cospoil}_u : \Pi\, u \Rightarrow \forall\, u$$

- For **cartesian** multipliers, we get

$$\Finv\, u = \Omega\, u \qquad \Pi\, u = \forall\, u$$

- For **cancellative and affine** multipliers (where $\Finv_U$ is **fully faithful**), we get invertible (co-)units

$$\mathsf{const}_u : 1 \cong \forall\, u \circ \Finv\, u$$

$$\mathsf{unmer}_u : \forall\, u \circ \between\, u \cong 1$$

- For **cancellative, affine and connection-free** multipliers, we get Φ/extent.

Quantification Theorem

- For **semicartesian** multipliers (where $\Omega\, u$ exists), we get

$$\mathsf{spoil}_u : \Finv\, u \Rightarrow \Omega\, u \qquad \mathsf{cospoil}_u : \Pi\, u \Rightarrow \forall\, u$$

- For **cartesian** multipliers, we get

$$\Finv\, u = \Omega\, u \qquad \Pi\, u = \forall\, u$$

- For **cancellative and affine** multipliers (where $\Finv_U$ is **fully faithful**), we get invertible (co-)units

$$\mathsf{const}_u : 1 \cong \forall\, u \circ \Finv\, u$$
$$\mathsf{unmer}_u : \forall\, u \circ \between\, u \cong 1$$

- For **cancellative, affine and connection-free** multipliers, we get Φ/extent.

Quantification Theorem

- For **semicartesian** multipliers (where $\Omega\, u$ exists), we get

$$\mathsf{spoil}_u : Ⅎ\, u \Rightarrow \Omega\, u \qquad \mathsf{cospoil}_u : \Pi\, u \Rightarrow \forall\, u$$

- For **cartesian** multipliers, we get

$$Ⅎ\, u = \Omega\, u \qquad \Pi\, u = \forall\, u$$

- For **cancellative and affine** multipliers
  (where $Ⅎ_U$ is **fully faithful**), we get invertible (co-)units

$$\mathsf{const}_u : 1 \cong \forall\, u \circ Ⅎ\, u$$
$$\mathsf{unmer}_u : \forall\, u \circ \between\, u \cong 1$$

- For **cancellative, affine and connection-free** multipliers,
  we get Φ/extent.

Definition

$\mathbb{U}$ is **atomic** if $\mathbb{U} \to -$ has a right adjoint $\sqrt[\mathbb{U}]{-}$.

Theorem

…iff $\Pi(u : \mathbb{U})$ has a right adjoint $\between u$.
(∇ in Yet87)

Theorem

Representables (e.g. $\mathbb{I}$) are atomic.

If $\mathbb{U}$ is atomic, then

$$
\begin{array}{lllll}
\Gamma & \vdash & \Pi\, u.(A_1\, u \uplus A_2\, u) & \to & (\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u) \\
 & & & \Updownarrow & \\
\Gamma, u : \mathbb{U} & \vdash & (A_1\, u \uplus A_2\, u) & \to & \between u.(\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u) \\
 & & & \Updownarrow & \\
\Gamma, u : \mathbb{U} & \vdash_{i=1,2} & A_i\, u & \to & \between u.(\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u) \\
 & & & \Updownarrow & \\
\Gamma & \vdash_{i=1,2} & \mathrm{inj}_i : \quad \Pi\, u.A_i\, u & \to & (\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u)
\end{array}
$$

Counterexample

Bool is not atomic.

Higher-Dimensional Pattern Matching

Definition

$\mathbb{U}$ is **atomic** if $\mathbb{U} \to -$ has a right adjoint $\sqrt[\mathbb{U}]{-}$.

Theorem

...iff $\Pi(u : \mathbb{U})$ has a right adjoint $\between u$.
(∇ in Yet87)

Theorem

Representables (e.g. $\mathbb{I}$) are atomic.

If $\mathbb{U}$ is atomic, then

$$
\begin{array}{lllcl}
\Gamma & \vdash & \Pi\, u.(A_1\, u \uplus A_2\, u) & \to & (\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u) \\
 & & & \Updownarrow & \\
\Gamma, u : \mathbb{U} & \vdash & (A_1\, u \uplus A_2\, u) & \to & \between u.(\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u) \\
 & & & \Updownarrow & \\
\Gamma, u : \mathbb{U} & \vdash_{i=1,2} & A_i\, u & \to & \between u.(\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u) \\
 & & & \Updownarrow & \\
\Gamma & \vdash_{i=1,2} & \mathrm{inj}_i : \Pi\, u.A_i\, u & \to & (\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u)
\end{array}
$$

Counterexample

Bool is not atomic.

Higher-Dimensional Pattern Matching

Definition

$\mathbb{U}$ is **atomic** if $\mathbb{U} \to -$ has a right adjoint $\sqrt[\mathbb{U}]{-}$.

Theorem

... iff $\Pi(u : \mathbb{U})$ has a right adjoint $\between u$.
(∇ in Yet87)

Theorem

Representables (e.g. $\mathbb{I}$) are atomic.

If $\mathbb{U}$ is atomic, then

$$\begin{array}{llrcl}
\Gamma & \vdash & \Pi\, u.(A_1\, u \uplus A_2\, u) & \to & (\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u) \\
 & & & \Updownarrow & \\
\Gamma, u : \mathbb{U} & \vdash & (A_1\, u \uplus A_2\, u) & \to & \between\, u.(\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u) \\
 & & & \Updownarrow & \\
\Gamma, u : \mathbb{U} & \vdash_{i=1,2} & A_i\, u & \to & \between\, u.(\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u) \\
 & & & \Updownarrow & \\
\Gamma & \vdash_{i=1,2} & \mathrm{inj}_i : \quad \Pi\, u.A_i\, u & \to & (\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u)
\end{array}$$

Counterexample

Bool is not atomic.

Higher-Dimensional Pattern Matching

Definition

$\mathbb{U}$ is **atomic** if $\mathbb{U} \to -$ has a right adjoint $\sqrt[\mathbb{U}]{-}$.

Theorem

... iff $\Pi(u : \mathbb{U})$ has a right adjoint $\between u$.
(∇ in Yet87)

Theorem

Representables (e.g. $\mathbb{I}$) are atomic.

If $\mathbb{U}$ is atomic, then

$$
\begin{array}{llrcl}
\Gamma & \vdash & \Pi\, u.(A_1\, u \uplus A_2\, u) & \to & (\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u) \\
 & & & \Updownarrow & \\
\Gamma, u : \mathbb{U} & \vdash & (A_1\, u \uplus A_2\, u) & \to & \between\, u.(\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u) \\
 & & & \Updownarrow & \\
\Gamma, u : \mathbb{U} & \vdash_{i=1,2} & A_i\, u & \to & \between\, u.(\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u) \\
 & & & \Updownarrow & \\
\Gamma & \vdash_{i=1,2} & \mathrm{inj}_i : \quad \Pi\, u.A_i\, u & \to & (\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u)
\end{array}
$$

Counterexample

Bool is not atomic.

Higher-Dimensional Pattern Matching

Definition

$\mathbb{U}$ is **atomic** if $\mathbb{U} \to -$ has a right adjoint $\sqrt[\mathbb{U}]{-}$.

Theorem

…iff $\Pi(u : \mathbb{U})$ has a right adjoint $\between u$.
(∇ in Yet87)

Theorem

Representables (e.g. $\mathbb{I}$) are atomic.

If $\mathbb{U}$ is atomic, then

$$
\begin{array}{llrcl}
\Gamma & \vdash & \Pi\, u.(A_1\, u \uplus A_2\, u) & \to & (\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u) \\
 & & & \Updownarrow & \\
\Gamma, u : \mathbb{U} & \vdash & (A_1\, u \uplus A_2\, u) & \to & \between u.(\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u) \\
 & & & \Updownarrow & \\
\Gamma, u : \mathbb{U} & \vdash_{i=1,2} & A_i\, u & \to & \between u.(\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u) \\
 & & & \Updownarrow & \\
\Gamma & \vdash_{i=1,2} & \mathrm{inj}_i : \quad \Pi\, u.A_i\, u & \to & (\Pi\, u.A_1\, u) \uplus (\Pi\, u.A_2\, u)
\end{array}
$$

Counterexample

Bool is not atomic.

Examples

Constructor for ${}^{\Omega u \dashv}\Pi\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \Omega\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \mathrm{mod}_{\Pi u}\, a : \langle \Pi\, u \mid A \rangle}$$

Projection by co-unit:
$\mathrm{app}_{v/u} : (\Omega\, v \shortmid \langle \Pi\, u \mid A \rangle) \to A[\mathrm{copy}_{v/u}\uparrow]$

Constructor for ${}^{\Finv u \dashv}\forall\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \Finv\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \mathrm{mod}_{\forall u}\, a : \langle \forall\, u \mid A \rangle}$$

Projection by co-unit:
$\mathrm{app}_{v/u} : (\Finv\, v \shortmid \langle \forall\, u \mid A \rangle) \to A[\mathrm{copy}_{v/u}\uparrow]$

Constructor for ${}^{\forall u \dashv}\between\, u$:

$$\frac{\Xi \mid \forall\, u(\Gamma) \vdash a : A}{\Xi, u : \mathbb{U} \mid \Gamma \vdash \mathrm{mod}_{\between u}\, a : \langle \between\, u \mid A \rangle}$$

Projection by co-unit:
$\mathrm{unmer}_u : (\forall\, u \shortmid \langle \between\, u \mid A \rangle) \to A[\mathrm{const}_u\uparrow]$

Examples

Constructor for ${}^{\Omega\, u \dashv} \Pi\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \Omega\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \lambda_u\, a : \langle \Pi\, u \mid A \rangle}$$

Projection by co-unit:
$\text{app}_{v/u} : (\Omega\, v \shortmid \langle \Pi\, u \mid A \rangle) \to A[\text{copy}_{v/u} \uparrow]$

Constructor for ${}^{\exists\, u \dashv} \forall\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \exists\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \text{mod}_{\forall\, u}\, a : \langle \forall\, u \mid A \rangle}$$

Projection by co-unit:
$\text{app}_{v/u} : (\exists\, v \shortmid \langle \forall\, u \mid A \rangle) \to A[\text{copy}_{v/u} \uparrow]$

Constructor for ${}^{\forall\, u \dashv} \between\, u$:

$$\frac{\Xi \mid \forall\, u(\Gamma) \vdash a : A}{\Xi, u : \mathbb{U} \mid \Gamma \vdash \text{mod}_{\between\, u}\, a : \langle \between\, u \mid A \rangle}$$

Projection by co-unit:
$\text{unmer}_u : (\forall\, u \shortmid \langle \between\, u \mid A \rangle) \to A[\text{const}_u \uparrow]$

Examples

Constructor for ${}^{\Omega\, u \dashv}\Pi\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \Omega\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \lambda_u\, a : \langle \Pi\, u \mid A\rangle}$$

Projection by co-unit:
$\mathrm{app}_{v/u} : (\Omega\, v \shortmid \langle \Pi\, u \mid A\rangle) \to A[\mathrm{copy}_{v/u}\uparrow]$

Constructor for ${}^{\Finv\, u \dashv}\forall\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \Finv\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \mathrm{mod}_{\forall\, u}\, a : \langle \forall\, u \mid A\rangle}$$

Projection by co-unit:
$\mathrm{app}_{v/u} : (\Finv\, v \shortmid \langle \forall\, u \mid A\rangle) \to A[\mathrm{copy}_{v/u}\uparrow]$

Constructor for ${}^{\forall\, u \dashv}\between\, u$:

$$\frac{\Xi \mid \forall\, u(\Gamma) \vdash a : A}{\Xi, u : \mathbb{U} \mid \Gamma \vdash \mathrm{mod}_{\between\, u}\, a : \langle \between\, u \mid A\rangle}$$

Projection by co-unit:
$\mathrm{unmer}_u : (\forall\, u \shortmid \langle \between\, u \mid A\rangle) \to A[\mathrm{const}_u\uparrow]$

Examples

Constructor for ${}^{\Omega u \dashv}\Pi\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \Omega\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \lambda_u\, a : \langle \Pi\, u \mid A \rangle}$$

Projection by co-unit:
$\mathrm{app}_{v/u} : (\Omega\, v \shortmid \langle \Pi\, u \mid A \rangle) \to A[\mathrm{copy}_{v/u}\uparrow]$

Constructor for ${}^{\exists u \dashv}\forall\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \exists\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \mathrm{mod}_{\forall u}\, a : \langle \forall\, u \mid A \rangle}$$

Projection by co-unit:
$\mathrm{app}_{v/u} : (\exists\, v \shortmid \langle \forall\, u \mid A \rangle) \to A[\mathrm{copy}_{v/u}\uparrow]$

Constructor for ${}^{\forall u \dashv}\between\, u$:

$$\frac{\Xi \mid \forall\, u(\Gamma) \vdash a : A}{\Xi, u : \mathbb{U} \mid \Gamma \vdash \mathrm{mod}_{\between u}\, a : \langle \between\, u \mid A \rangle}$$

Projection by co-unit:
$\mathrm{unmer}_u : (\forall\, u \shortmid \langle \between\, u \mid A \rangle) \to A[\mathrm{const}_u\uparrow]$

Examples

Constructor for $^{\Omega u \dashv}\Pi\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \Omega\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \lambda_u\, a : \langle \Pi\, u \mid A \rangle}$$

Projection by co-unit:
$\mathrm{app}_{v/u} : (\Omega\, v \shortmid \langle \Pi\, u \mid A \rangle) \to A[\mathrm{copy}_{v/u}\uparrow]$

Constructor for $^{ⅎ u \dashv}\forall\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid ⅎ\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \Lambda_u\, a : \langle \forall\, u \mid A \rangle}$$

Projection by co-unit:
$\mathrm{app}_{v/u} : (ⅎ\, v \shortmid \langle \forall\, u \mid A \rangle) \to A[\mathrm{copy}_{v/u}\uparrow]$

Constructor for $^{\forall u \dashv}≬\, u$:

$$\frac{\Xi \mid \forall\, u(\Gamma) \vdash a : A}{\Xi, u : \mathbb{U} \mid \Gamma \vdash \mathrm{mod}_{≬ u}\, a : \langle ≬\, u \mid A \rangle}$$

Projection by co-unit:
$\mathrm{unmer}_u : (\forall\, u \shortmid \langle ≬\, u \mid A \rangle) \to A[\mathrm{const}_u\uparrow]$

Examples

Constructor for ${}^{\Omega u \dashv}\Pi\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \Omega\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \lambda_u\, a : \langle \Pi\, u \mid A \rangle}$$

Projection by co-unit:
$\mathsf{app}_{v/u} : (\Omega\, v \shortmid \langle \Pi\, u \mid A \rangle) \to A[\mathsf{copy}_{v/u}\!\uparrow]$

Constructor for ${}^{\reflectbox{$\vdash$} u \dashv}\forall\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \reflectbox{$\vdash$}\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \Lambda_u\, a : \langle \forall\, u \mid A \rangle}$$

Projection by co-unit:
$\mathsf{app}_{v/u} : (\reflectbox{$\vdash$}\, v \shortmid \langle \forall\, u \mid A \rangle) \to A[\mathsf{copy}_{v/u}\!\uparrow]$

Constructor for ${}^{\forall u \dashv}\between\, u$:

$$\frac{\Xi \mid \forall\, u(\Gamma) \vdash a : A}{\Xi, u : \mathbb{U} \mid \Gamma \vdash \mathsf{mod}_{\between u}\, a : \langle \between\, u \mid A \rangle}$$

Projection by co-unit:
$\mathsf{unmer}_u : (\forall\, u \shortmid \langle \between\, u \mid A \rangle) \to A[\mathsf{const}_u\!\uparrow]$

Examples

Constructor for ${}^{\Omega u \dashv}\Pi\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \Omega\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \lambda_u\, a : \langle \Pi\, u \mid A \rangle}$$

Projection by co-unit:
$\mathsf{app}_{v/u} : (\Omega\, v \shortmid \langle \Pi\, u \mid A \rangle) \to A[\mathsf{copy}_{v/u}\uparrow]$

Constructor for ${}^{\exists u \dashv}\forall\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \exists\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \Lambda_u\, a : \langle \forall\, u \mid A \rangle}$$

Projection by co-unit:
$\mathsf{app}_{v/u} : (\exists\, v \shortmid \langle \forall\, u \mid A \rangle) \to A[\mathsf{copy}_{v/u}\uparrow]$

Constructor for ${}^{\forall u \dashv}\between\, u$:

$$\frac{\Xi \mid \forall\, u(\Gamma) \vdash a : A}{\Xi, u : \mathbb{U} \mid \Gamma \vdash \mathsf{mod}_{\between u}\, a : \langle \between\, u \mid A \rangle}$$

Projection by co-unit:
$\mathsf{unmer}_u : (\forall\, u \shortmid \langle \between\, u \mid A \rangle) \to A[\mathsf{const}_u\uparrow]$

Examples

Constructor for ${}^{\Omega u \dashv}\Pi\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \Omega\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \lambda_u\, a : \langle \Pi\, u \mid A \rangle}$$

Projection by co-unit:
$\mathsf{app}_{v/u} : (\Omega\, v \mathrel{\text{ı}} \langle \Pi\, u \mid A \rangle) \to A[\mathsf{copy}_{v/u}\uparrow]$

Constructor for ${}^{\text{Ⅎ} u \dashv}\forall\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \text{Ⅎ}\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \Lambda_u\, a : \langle \forall\, u \mid A \rangle}$$

Projection by co-unit:
$\mathsf{app}_{v/u} : (\text{Ⅎ}\, v \mathrel{\text{ı}} \langle \forall\, u \mid A \rangle) \to A[\mathsf{copy}_{v/u}\uparrow]$

Constructor for ${}^{\forall u \dashv}\between\, u$:

$$\frac{\Xi \mid \forall\, u(\Gamma) \vdash a : A}{\Xi, u : \mathbb{U} \mid \Gamma \vdash \mathsf{mer}_u\, a : \langle \between\, u \mid A \rangle}$$

Projection by co-unit:
$\mathsf{unmer}_u : (\forall\, u \mathrel{\text{ı}} \langle \between\, u \mid A \rangle) \to A[\mathsf{const}_u\uparrow]$

Examples

Constructor for ${}^{\Omega\, u \dashv}\Pi\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \Omega\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \lambda_u\, a : \langle \Pi\, u \mid A \rangle}$$

Projection by co-unit:
$\mathrm{app}_{v/u} : (\Omega\, v \shortmid \langle \Pi\, u \mid A \rangle) \to A[\mathrm{copy}_{v/u}{\uparrow}]$

Constructor for ${}^{\text{Ⅎ}\, u \dashv}\forall\, u$:

$$\frac{\Xi, u : \mathbb{U} \mid \text{Ⅎ}\, u(\Gamma) \vdash a : A}{\Xi \mid \Gamma \vdash \Lambda_u\, a : \langle \forall\, u \mid A \rangle}$$

Projection by co-unit:
$\mathrm{app}_{v/u} : (\text{Ⅎ}\, v \shortmid \langle \forall\, u \mid A \rangle) \to A[\mathrm{copy}_{v/u}{\uparrow}]$

Constructor for ${}^{\forall\, u \dashv}\between\, u$:

$$\frac{\Xi \mid \forall\, u(\Gamma) \vdash a : A}{\Xi, u : \mathbb{U} \mid \Gamma \vdash \mathrm{mer}_u\, a : \langle \between\, u \mid A \rangle}$$

Projection by co-unit:
$\mathrm{unmer}_u : (\forall\, u \shortmid \langle \between\, u \mid A \rangle) \to A[\mathrm{const}_u{\uparrow}]$