I wanted to try figuring out a sort of internal language corresponding to the double category **Rel**.

You have a calculus of "values" corresponding to one edge of **Rel** and a sort of relational calculus of "predicates" corresponding to the other edge. Squares ought to correspond to judgements stating a value satisfies a predicate.

The language of "values" handling Cartesian product of sets has product types (in category theory terms is Cartesian.)

The language of "predicates" ought to be more complicated. **Rel** is a closed monoidal category over Cartesian product of sets. One has an isomorphism $\mathbf{Rel}(A, B \otimes C) \sim \mathbf{Rel}(A \otimes B, C)$. So some sort of linear type theory is required.

This is the core framework. I've been thinking about further extensions but I want to see how far I can get characterizing **Rel** with as few language features as possible. Later on I give a few possible extensions.

Core Calculi

Types	$t ::= \top \mid t \times t$
Values	$v ::= x \mid \mathbf{tt} \mid \mathbf{fst}(v) \mid \mathbf{snd}(v) \mid (v, v)$
Predicates	$p ::= x \mid \mathbf{pass}(p, p) \mid p\,p \mid \forall(x\colon t).p$
Environment	$\Gamma ::= \cdot \mid \Gamma,\, x\colon t$
Substitutions	$\sigma ::= \cdot \mid \sigma,\, v \models x$

The core relational calculus is the linear lambda calculus modulo a few symbol changes. The core value calculus has only product types.

I'm not really sure small steps semantics make sense with respect to the relational calculus but the value calculus corresponding to Set ought to have simple deterministic semantics.

Relational Calculus

$$\frac{\Gamma, x\colon t_0 \vdash p\colon t_1}{\Gamma \vdash \forall(x\colon t_0).p\colon t_0 \times t_1}\ (\times\text{-INTRO})$$

$$\frac{\Gamma \vdash p_0\colon t_0 \times t_1 \qquad \Delta \vdash p_1\colon t_1}{\Gamma,\, \Delta \vdash p_0 p_1 : t_1}\ (\times\text{-ELIM})$$

$$\frac{\Gamma \vdash p_0\colon \top \qquad \Delta \vdash p_1 : t}{\Gamma,\, \Delta \vdash \mathbf{pass}(p_0, p_1)\colon t}\ (\top\text{-ELIM})$$

$$\frac{\Gamma \vdash p_0\colon \top \qquad \Delta \vdash p_1 : t}{\Gamma,\, \Delta \vdash \mathbf{pass}(p_0, p_1)\colon t}\ (\top\text{-ELIM})$$

$$\frac{}{(\forall(x\colon t).p_0)\, p_1 \to [x := p_1]p_0}\ (\times\text{-STEP})$$

Value Calculus

$$\frac{}{\Gamma \vdash \mathbf{tt}\colon \top}\ (\top\text{-INTRO})$$

$$\frac{\Gamma \vdash v\colon t_0 \times t_1}{\Gamma \vdash \mathbf{fst}(v)\colon t_0}\ (\times\text{-ELIM-1})$$

$$\frac{\Gamma \vdash v\colon t_0 \times t_1}{\Gamma \vdash \mathbf{snd}(v)\colon t_1}\ (\times\text{-ELIM-2})$$

$$\frac{\Gamma \vdash v_0\colon t_0 \qquad \Gamma \vdash v_1 : t_1}{\Gamma \vdash (v_0, v_1)\colon t_0 \times t_1}\ (\times\text{-INTRO})$$

$$\frac{}{\mathbf{fst}(v_0, v_1) \to v_0}\ (\times\text{-STEP-1})$$

$$\frac{}{\mathbf{snd}(v_0, v_1) \to v_1}\ (\times\text{-STEP-2})$$

Satisfaction Judgements

I need a better symbol here.

$$\frac{v_0 \to v_1 \qquad v_0 \models p \quad [\sigma]}{v_1 \models p \quad [\sigma]}\ (\text{v-STEP-SAT})$$

$$\frac{p_0 \to p_1 \qquad v \models p_0 \quad [\sigma]}{v \models p_1 \quad [\sigma]}\ (\text{p-STEP-SAT})$$

$$\frac{v \models \mathbf{pass}(p_0, p_1) \quad [\sigma]}{\mathbf{tt} \models p_0 \quad [\sigma]}\ (\top\text{-SAT})$$

$$\frac{v_1 \models p \quad [\sigma,\, v_0 \models x]}{(v_0, v_1) \models \forall(x\colon t).p \quad [\sigma]}\ (\times\text{-SAT})$$

Examples

Pattern matching on equality

$(v, v) \models \forall(x\colon t).x$

Transposition

$\forall(p\colon t \times t \times \top)(x\colon t)(y\colon t).p\,y\,x$

Disjoint Unions

Disjoint unions in set become Cartesian product/coproduct in **Rel**.

I am fairly confident in a simple extension to disjoint unions of sets which are sum types in the value calculus and product types in the predicate calculus.

Types	$t ::= \ldots \mid \bot \mid t + t$
Values	$v ::= \ldots \mid \mathbf{absurd}_t(v) \mid \mathbf{i1}_t(v) \mid \mathbf{i2}_t(v) \mid \mathbf{mtc}\ v\,\{x \mapsto v; x \mapsto v\}$
Predicates	$p ::= \ldots \mid \mathbf{false} \mid \mathbf{left}(p) \mid \mathbf{right}(p) \mid [p; p]$

Values

$$\frac{\Gamma \vdash v\colon \bot}{\Gamma \vdash \mathbf{absurd}_t(v)\colon t}\ (\bot\text{-ELIM})$$

$$\frac{\Gamma \vdash v\colon t_0}{\Gamma \vdash \mathbf{i1}_{t_1}(v)\colon t_0 + t_1}\ (+\text{-INTRO-1})$$

$$\frac{\Gamma \vdash v\colon t_1}{\Gamma \vdash \mathbf{i2}_{t_0}(v)\colon t_0 + t_1}\ (+\text{-INTRO-2})$$

$$\frac{\Gamma \vdash v_0\colon t_0 + t_1 \qquad \Gamma,\, x_0\colon t_0 \vdash v_1 : t_2 \qquad \Gamma,\, x_1\colon t_1 \vdash v_1 : t_2}{\Gamma \vdash \mathbf{mtc}\ v_0\,\{x_0 \mapsto v_1; x_1 \mapsto v_2\}\colon t_2}\ ($$

$$\frac{}{\mathbf{mtc}\ \mathbf{i1}_t(v_0)\,\{x_0 \mapsto v_1; x_1 \mapsto v_2\} \to [x_0 := v_0]v_1}\ (+\text{-STEP-1})$$

$$\frac{}{\mathbf{mtc}\ \mathbf{i2}_t(v_0)\,\{x_0 \mapsto v_1; x_1 \mapsto v_2\} \to [x_1 := v_0]v_2}\ (+\text{-STEP-2})$$

Predicates

$$\frac{}{\Gamma \vdash \mathbf{false} \colon \bot}\ (\bot\text{-INTRO})$$

$$\frac{\Gamma \vdash p \colon t_0 + t_1}{\Gamma \vdash \mathbf{left}(p) \colon t_0}\ (+\text{-ELIM-1})$$

$$\frac{\Gamma \vdash p \colon t_0 + t_1}{\Gamma \vdash \mathbf{left}(p) \colon t_1}\ (+\text{-ELIM-2})$$

$$\frac{\Gamma \vdash p_0 \colon t_0 \qquad \Gamma \vdash p_1 \colon t_1}{\Gamma \vdash [p_0; p_1] \colon t_0 + t_1}\ (+\text{-INTRO})$$

$$\frac{}{\mathbf{left}([p_0; p_1]) \to p_0}\ (+\text{-STEP-1})$$

$$\frac{}{\mathbf{right}([p_0; p_1]) \to p_1}\ (+\text{-STEP-2})$$

Satisfies

$$\frac{v \models p_0 \quad [\sigma]}{\mathbf{i1}_t(v) \models [p_0; p_1] \quad [\sigma]}\ (+\text{-SAT-1})$$

$$\frac{v \models p_1 \quad [\sigma]}{\mathbf{i2}_t(v) \models [p_0; p_1] \quad [\sigma]}\ (+\text{-SAT-2})$$

$$\frac{\mathbf{absurd}_t(v) \models p \quad [\sigma]}{v \models \mathbf{false} \quad [\sigma]}\ (\bot\text{-SAT})$$

Dependent Sums

If product of sets becomes an internal hom in the predicate calculus then dependent sums ought to become a little like Π types. So the predicate calculus effectively becomes like a linear System-F.

Some things become awkward to interpret here though.

I also really can't figure out unpacking. It's messy if you don't want full dependent types.

Types	$t ::= \ldots \mid x \mid * \mid \mathbf{head}(v) \mid \Sigma(x\colon\ *).t$
Values	$v ::= \ldots \mid \mathbf{tail}(v) \mid \langle x := t, v\rangle$
Predicates	$p ::= \ldots \mid p\,t \mid \mathbf{M}(x\colon\ *).p$
Substitutions	$\sigma ::= \ldots \mid \sigma,\, t \models x$

Not really good at the typing judgements for dependent sum types.

Values

$$\frac{\Gamma \vdash v \colon \Sigma(x\colon\ *).t}{\Gamma \vdash \mathbf{head}(v) \colon\ *}\ (\Sigma\text{-ELIM-1})$$

$$\frac{\Gamma \vdash v \colon \Sigma(x\colon\ *).t}{\Gamma \vdash \mathbf{tail}(v) \colon [x := \mathbf{head}(v)]t}\ (\Sigma\text{-ELIM-2})$$

$$\frac{\Gamma \vdash t_0 \colon\ * \qquad \Gamma,\, x\colon\ * \vdash v \colon t_1}{\Gamma \vdash \langle x := t_0, v\rangle \colon \Sigma(x\colon\ *).t_0}\ (\Sigma\text{-INTRO})$$

Predicates

$$\frac{\Gamma \vdash p \colon \Sigma(x\colon\ *).t_1 \qquad \Delta \vdash t_0 \colon\ *}{\Gamma,\, \Delta \vdash p_0\, t_0 \colon [x := t_0]t_1}\ (\Sigma\text{-ELIM})$$

$$\frac{\Gamma,\, x\colon\ * \vdash p \colon t}{\Gamma \vdash \mathbf{M}(x\colon\ *).p \colon \Sigma(x\colon\ *).t}\ (\Sigma\text{-INTRO})$$

$$\frac{}{(\mathbf{M}(x\colon\ *).p)\, t \to [x := t]p}\ (\Sigma\text{-STEP})$$

Satisfaction

I can't figure out satisfaction at all.

$$\frac{v \models p \quad [\sigma,\, t \models x]}{\langle x := t, v\rangle \models \mathbf{M}(x\colon\ *).p \quad [\sigma]}\ (\Sigma\text{-SAT})$$

Categorical Semantics

The intent is to create calculi encoding the core features of the double category **Rel**.

If this is successful then terms and types ought to map to **Rel**as follows.

$$\begin{tikzcd}
\top \arrow[rr, "\sigma_v"] \arrow[d, "\sigma_p"'] & \arrow[d, phantom, "\Downarrow\, v\models p\,[\sigma_v,\sigma_p]"] & \times_{x\colon t\in\Gamma_v} t \arrow[d, "v"] \\
\times_{x\colon t\in\Gamma_p} t \arrow[rr, "p"'] & & t
\end{tikzcd}$$

The Future?

Satisifies judgments correspond to thin squares. Moving to more generally categories such as **Span** or **Prof** or **Vect** for matrix math requires an interpretation of squares carrying constructive content.