logic in color

#3: Generalizing Logic I

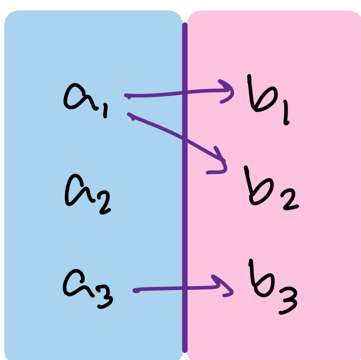
[Matrices & Monads]

Welcome back!

We've been thinking about relations.

A relation is a matrix of truth values.

$$A \xrightarrow{R} B$$



R	b_1	b_2	b_3
a_1	1	1	0
a_2	0	0	0
a_3	0	0	1

Composition is matrix multiplication

$$\Sigma_b R_{ab} \cdot S_{bc} \quad := \quad \exists b.\ aRb \wedge bSc$$

& identity is the identity matrix.

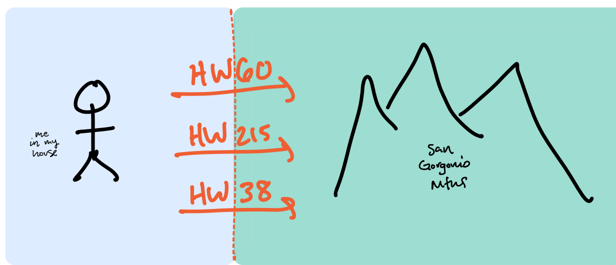
✱ Yet there are many kinds of data which can connect objects. ✱

This is the key to generalize logic.

A judgement can contain rich data, beyond just 0s + 1s:

it could be a distance

or a set of connections, & much more.



What kinds of judgements can we make?

cost/resources, action/input, a story, the environment

Type Theory began with realizing "whoa, judgements can be more than propositions."
Same here, but we just call it Logic.

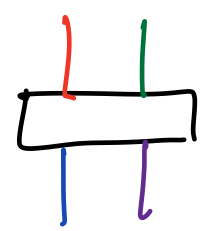
The only condition is that the data compose:
as long as we have "matrix multiplication"

$$a \; R{\circ}S \; c \;=\; \sum b{:}B.\; aRb \cdot bSc \qquad [?]^{\text{sets}}_{\text{dist}}$$

then the "kind of judgement" (which we'll call $\mathbb{V}$) likely forms its own kind of "$\mathbb{V}$-logic"!

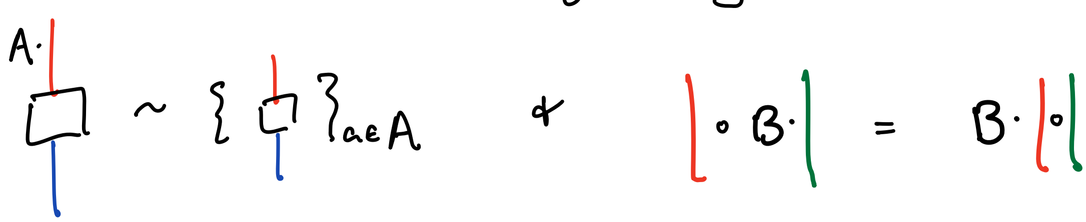
→ just one thing: What is the "kind of inference"; what kind of structure is $\mathbb{V}$?

Well, for "a notion of judgement & inference" $\mathbb{V}$ can be a double category with just one type & one term. This is a **monoidal category**.



What do we need for matrix multiplication?

(Set-indexed) sums, which get along with composition.



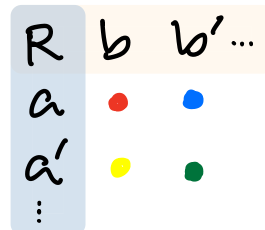
Let's call this condition being **tensored**.

def Let $\mathbb{V}$ be a tensored monoidal category.

Then Mat$\mathbb{V}$ is a double category:

type $\quad \boxed{A} \quad$ set

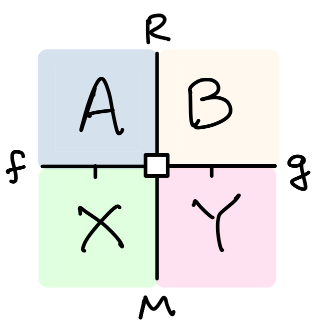
judgement $\quad \boxed{A \mid B} \quad$ matrix $R : A \times B \to \mathbb{V}$

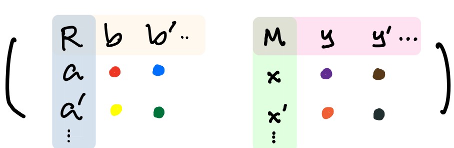


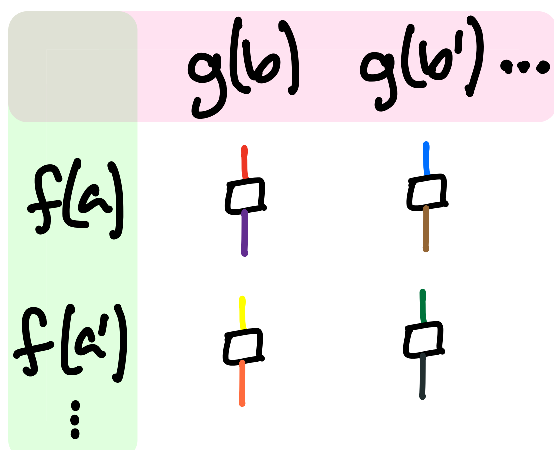
↳ composition $\quad a\,RS\,c = \sum b \in B.\ aRb \circ bSc$

↳ identity $\quad aIa' = \begin{cases} 1 & a = a' \\ 0 & a \neq a' \end{cases}$ $\quad \left(\begin{array}{l} 1 : \text{unique type} \\ 0 : \text{empty sum} \end{array} \right)$

inference







↳ sequence + parallel composition. $\quad$ ⧄

Mat$\mathbb{V}$ is the ground for "$\mathbb{V}$-valued logic."

what is it like for $\mathbb{V} = (\mathrm{Set}, \times, 1)$? $\quad (\mathbb{R}, \leq, +, 0)$?

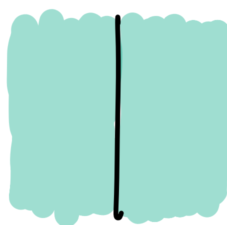
The world of $\mathrm{Mat}\mathbb{V}$ is nice,
but it's missing essential aspects of logic. (*)

The problem is that the data of judgements is not yet "in" the types — so far just sets. (ex)

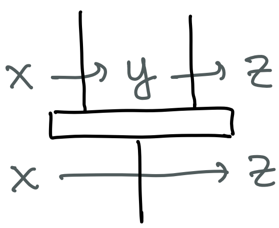
This is an issue even for relations:
plain old sets don't "know about" implication.

* what kind of structures do?

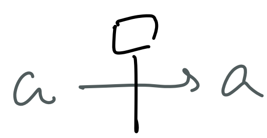
Let R be a relation on a set A.



$A : x \xrightarrow{R} y : A$



composition

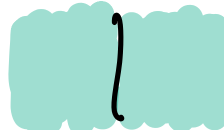


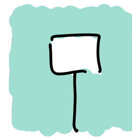
identity

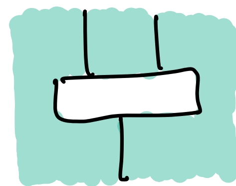
A preorder is a set & relation $A \xrightarrow{R} A$
with implications comp : $R \circ R \Rightarrow R$
& id : $1_A \Rightarrow R$.

This is a "monad" in Rel.

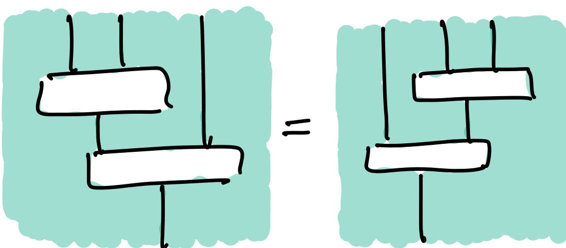
def Let $\mathbb{C}$ be a double category.

A **monad** in $\mathbb{C}$ is a judgement 

with inferences  unit

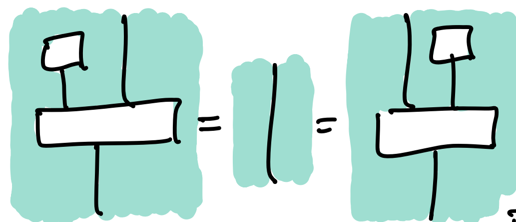
 join

so that



assoc

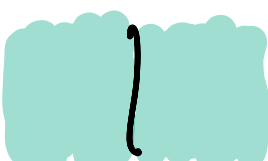
and

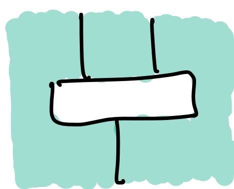


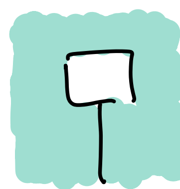
lunit runit

So a monad is "a judgement (string) with composition & identity"

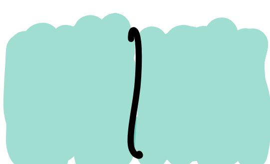
* What's a monad in Mat(Set)?

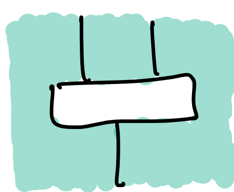


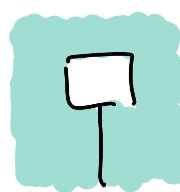




* What's a monad in Mat($\mathbb{R}$)?







For a double category $\mathbb{C}$ (with one condition)
there's a double category of

"monads & modules" in $\mathbb{C}$,
denoted $\mathrm{Mod}(\mathbb{C})$.

For $\mathbb{V}$-logic, we'll explore $\overline{\mathbb{V}} := \mathrm{Mod}(\mathrm{Mat}(\mathbb{V}))$.
This is a very rich world for generalized logic.

What can we learn & do in this world?

Questions / Thoughts ?

Thanks!