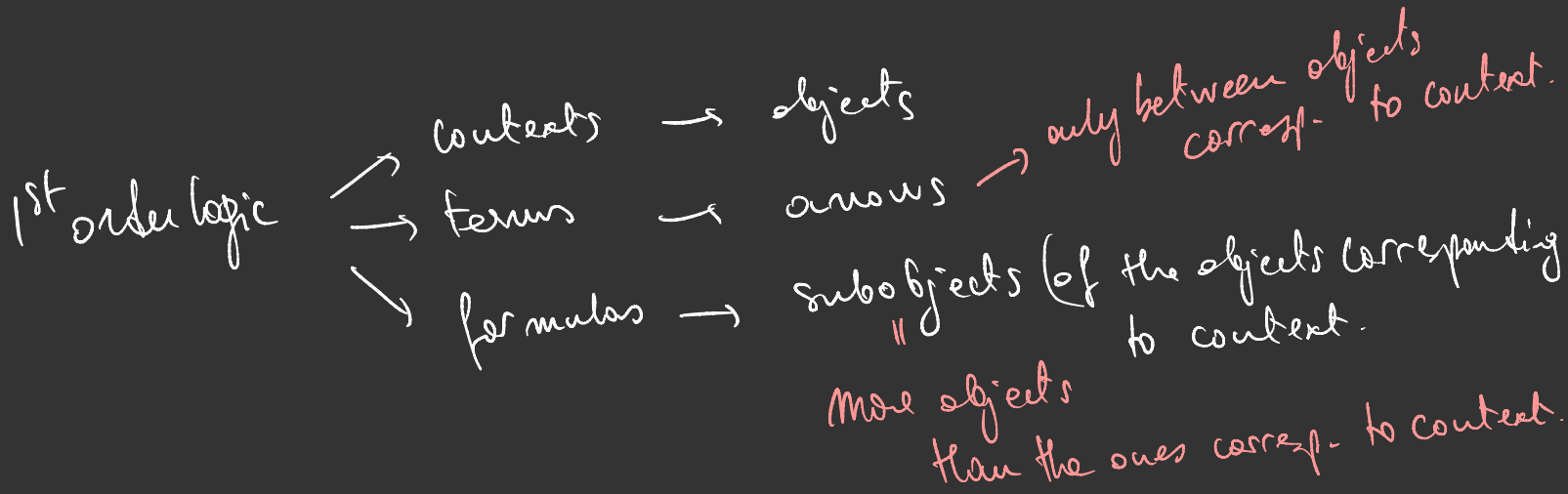


Syntactic cat for 1st order logic



need also arrows between the new objects arise to formula

Syntax

$\phi(x, y, z)$

$\psi(u, v)$

(x, y, z)
content

$(s(x, y, z), t(x, y, z))$

\longrightarrow

(u, v)
content

Sym(T)

$[\phi]$

$[\psi]$

$p \text{ is } \phi$

mono

$p \text{ is } \psi$
mono

$[x, y, z] \xrightarrow{[s, t]} [u, v]$
 $p \text{ is } (u=s \wedge t=v)$

in orange = new shows.

$[\phi] \xrightarrow{? \text{ arbitrary } p} [\psi]$

Syn(T)

objects = formula in context (\vec{x}, φ) (if $\varphi = T$ this is the object corresponding to the context)

morph. $(\vec{x}, \varphi) \xrightarrow{f} (\vec{y}, \psi)$ = "functional relations" in context (\vec{x}, \vec{y})

$f: (\vec{x}, \vec{y}, \rho)$ such that

• $\rho \vdash_{\vec{x}, \vec{y}} \varphi \wedge \psi$

• $\varphi \vdash_{\vec{x}} \exists \vec{y} \rho(\vec{x}, \vec{y})$

• $\rho(\vec{x}, \vec{y}), \rho(\vec{x}, \vec{y}') \vdash_{\vec{x}, \vec{y}, \vec{y}'} \vec{y} = \vec{y}'$

identity maps of (\vec{x}, ϕ) is $(\vec{x}, \vec{x}', \rho)$

$$\rho(x, x') \text{ is } x = x' \wedge \phi(x)$$

$$\left(\text{or } x = x' \wedge \phi(x) \wedge \phi(x') \right)$$

composition

$$(\vec{x}, \phi) \xrightarrow{(\vec{x}\vec{y}, \rho)} (\vec{y}, \psi) \xrightarrow{(\vec{y}\vec{z}, \lambda)} (\vec{z}, \zeta)$$

$$\underbrace{\hspace{10em}}_{(\vec{x}, \vec{z}, \mu)}$$

$$\mu(\vec{x}, \vec{z}) \text{ is } \exists \vec{y}, \rho(\vec{x}, \vec{y}) \wedge \lambda(\vec{y}, \vec{z}) \wedge \psi(y)$$

Remark 1: that the definition of $\text{Sgn}(T)$ uses

- the \wedge connector
- the \exists quantifier.

Hence it make sense only for regular theories T

Remark 2: if the theory has more constructors ($\circ, \cup, \Rightarrow, \neg, \forall$)
the definition of $\text{Sgn}(T)$ is the same, we simply get
more objects and more formulas.

Theorem : • if T is regular (coh, Heyt, Boolean), then
 $\text{Syn}(T)$ is a regular (coh, Heyt, or Boolean) category.
(details in Johnstone's Elephant D. 4.4)

• Soundness and completeness : $\text{Syn}(T)$ has a universal model \mathcal{U} of T
and for a formula φ $T \vdash \varphi \iff \mathcal{U} \models \varphi$
(φ derivable from axioms) φ is true in the universal model

• $(\text{Syn}(T), \mathcal{U})$ represents the functors of models of T

\mathcal{C} reg cat : $\text{Mod}(T, \mathcal{C}) \simeq_{\text{eq-cat}} \text{Fun}^{\text{reg}}(\text{Syn}(T), \mathcal{C})$

- Gödel completeness result for
 \mathcal{T} finitary classical first order theory
(Boolean) ($\top, \wedge, \perp, \vee, \exists, \forall, \Rightarrow, \neg$)

$$\mathcal{T} \vdash \varphi \iff M \models \varphi$$

φ is derivable from the axioms for any any model in Set

(topological interpretation in terms of topos by Deligne)

Beyond 1st order logic

the focus of 1st order logic is formula.

→ many operations on formulas $\neg, \wedge, \perp, \vee, \exists, \forall, \Rightarrow, \neg$

few operations on contexts / sort / types concatenation / products ^{cont.}

to go beyond: add more operations on sorts / types
constructor

- function types $X \rightarrow Y$ (λ -calculus)
- Σ, Π types (in DTT)

these theories beyond 1st order logic are also associated
to syntactic categories (Crole: Categories for Types)

But in general these theories do not have
completeness results (with respect to models in Set)

this is why 1st - ord. logic is so important: because it's
complete.

the Lawvere revolution

1) model theory logical th = language (syntax, proof theory)

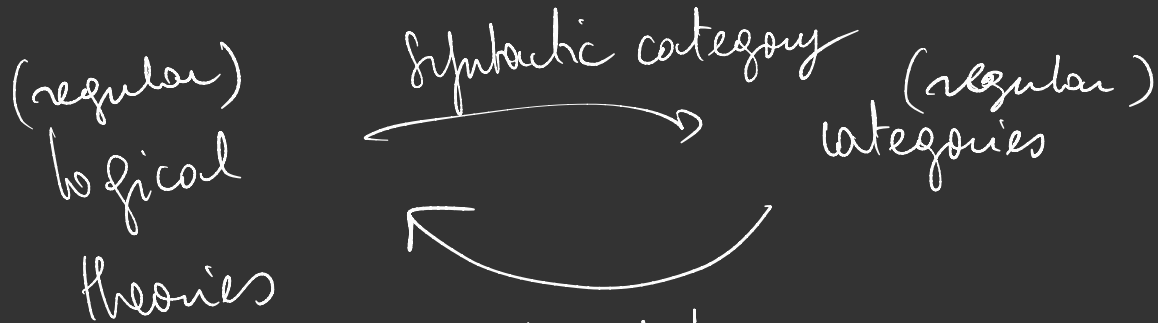
model: translation into another language
(e.g. \mathbb{Z} , \mathbb{F} ...)

2) Categorical Semantics language $\xrightarrow[\text{interpretation}]{\text{models}}$ in a category
heterogeneous

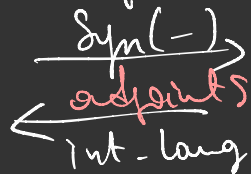
3) Lawvere: (syntactic) category $\xrightarrow[\text{functor}]{\text{models}}$ category

We've seen how to construct a category from a logical theory.

There is a process the opposite way:



lot of reg. theories



cat of reg cat

Example \mathcal{C} is a regular category.

can construct a regular theory.

Σ : $S =$ all objects of \mathcal{C} .

$F =$ all arrows of \mathcal{C} .

$R =$ all monomorphisms.

} \rightarrow $\left. \begin{array}{l} \text{Context} \\ \text{terms} \\ \text{formulas} \end{array} \right\}$
for regular logic.

the signature has a $\left. \begin{array}{l} \text{canonical} \\ \text{tautological} \end{array} \right\}$ model in \mathcal{C}

$A =$ axioms : all the sequent of Σ that are true in this model

(Σ, A) is the internal language of \mathcal{C} .