Programming Ontological Generators

Noah Chrein

April 6, 2019

Olog Golog

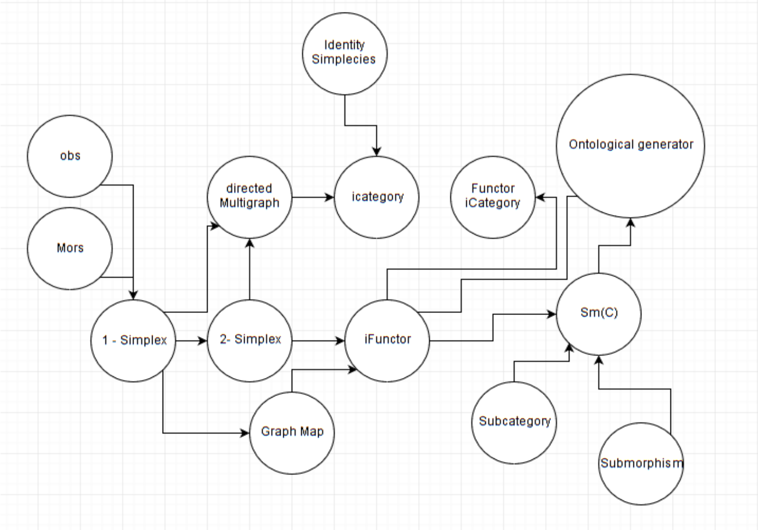
Identity Simplecies
obs
Mors
directed Multigraph
icategory
Functor iCategory
Ontological generator
1 - Simplex
2- Simplex
iFunctor
Sm(C)
Graph Map
Subcategory
Submorphism

Table of Contents

Directed Multigraph

A directed multigraph is two sets Ob, Mor, with two functions $dom, codom : Mor \to Ob$

-create object class

Directed Multigraph

A directed multigraph is two sets Ob, Mor, with two functions $dom, codom : Mor \to Ob$

-create object class
-create morphism class morphism.domain $=$ object,
morphism.codomain $=$ object

Directed Multigraph

A directed multigraph is two sets Ob, Mor, with two functions $dom, codom : Mor \to Ob$

-create object class
-create morphism class morphism.domain = object, morphism.codomain = object
-create multigraph class multigraph.morphisms = [morphisms] multigraph.objects = [objects]

Graph Maps

Graph Map

Let G, H be two graphs, A pair $F = (F^0 : ob(G) \to ob(H), F^1 : mor(G) \to mor(H))$ is a graph map, if $F : G \to H$ satisfies:

i)$F^1(f).dom = F^0(f.dom)$

ii)$F^1(f).codom = F^0(f.codom)$

G-Simplecies/ simplicial graphs

Consider the following multigraph Δ^2:
(tikz)

G-simplex

A G-simplex is a graph Map $F : \Delta^2 \to G$ which we write as $(f, g, gof) = (F(01), F(12), F(02))$
A the set of G-simplecies is GSimp, and comes with two maps$\pi_f, \pi_g : GSimp \to mor^2(G)$ by

$$\pi_f(F) = F_1(01) \text{ and } \pi_g(F) = F_1(12)$$

Consider a subset of G-simplecies $Simp \subseteq GSimp$
Simp is called coherant, if $\forall F \in Simp$, F is the only simplex such that $F(01) = f$ and $F(12) = g$

$$(\text{i.e. } |[\pi_f \times \pi_g]^{-1}(mor(G)^2)| \leq 1.)$$

simplicial graphs

simplicial graph

A simplicial graph C = (G,Simp) is a multigraph G with a coherant set of simplecies Simp.

By the uniqueness of Simp, when $(f, g, h) \in Simp$ we can define composition $g \circ f = h$

simplicial map

Let C, D be two simplicial graphs, a simplicial map is a double $(F, F^2) = (F^0, F^1, F^2)$, with F a graph Map and $F^2 : Simp_C \rightarrow Simp_D$ a function of sets such that the following equations hold:
i)$F_2(simp).ob = F_0(simp.ob)$
ii)$F_2(simp).mor = F_1(simp.mor)$

identities

Identity

Let C be a simplicial graph, a morphism $I : O \to O$ is an identity if $\forall f : A \to O, g : O \to B$ there is a simplex (f,I,f) and (I,g,g).

iCategory

an icategory is a simplicial graph such that every object o has an identity morphism $I(o) = I_o$, where $I : ob \to mor$

functors, subcategories

iFunctor

an iFunctor $F : C \to D$ between icategories, is a simplicial map such that:
$F^1(I(o)) = I(F^0(o))$
Q: is this redundant? Can this be derived via simplecies?
A: yes, the simplecies (I_o, I'_o, I_o) and (I_o, I'_o, I'_o) results in $I_o = I'_o$ by uniqueness. I.e. (defining two identities in an icategory will throw a uniqueness assertion error in Simp)

Sm(C)

subcategory

a subcategory $S \leq C$ is three subsets $obs_S \leq ob_C$, $mors_S \leq mor_C$, $simps_S \leq simp_C$ such that the inclusion induces a functor $i : S \to C$ (this means that the inclusion maps (i^1, i^2, i^3) satisfy the graph map, simplicial map and iFunctor conditions)

submorphism

Let $Hom_C(S, S') = \{f : o \to o' | o \in S, o' \in S'\}$
A submorphism $\mathcal{F} : S \to S'$ is a subset $\mathcal{F} \subseteq Hom_C(S, S')$ (i.e. some arbitrary collection of maps between objects of either subcategory

Sm(C)

Sm(C)

Let Sm(C) be the icategory of subcategories and submorphisms.

of course we have to define the simplecies (and identities):
let

$$\mathcal{G} \circ \mathcal{F} = \{g \circ f | f : o \to o', g : o' \to o'' g.dom = f.codom\}$$

and define the simplecies $(\mathcal{F}, \mathcal{G}, \mathcal{G} \circ \mathcal{F})$ then,
$I_S = \{Id_o | o \in S\}$

Ontological Generators

Ontological Generator

An ontological generator is a Functor $OG : C \to sm(C)$

computational ontologies

-git
-variable instantiating
-file systems

ontologies in math

- Topological Ontology.
- semantic / logical ontology

math for ontologies

-Sheaf theory / $OG_{\$}$ makes $\mathbb{C}$ into a <u>Site</u>
-deduction as gluing property

ontologies for ontologies

-Ontological Readers
-Ontological Updaters / Actors
-Axiom objects
-Understanding as Axiomatic covering by generators