Diary of a software engineer using categories

Angeline Aguinaldo
July 6, 2021
Berkeley Seminar, Topos Institute

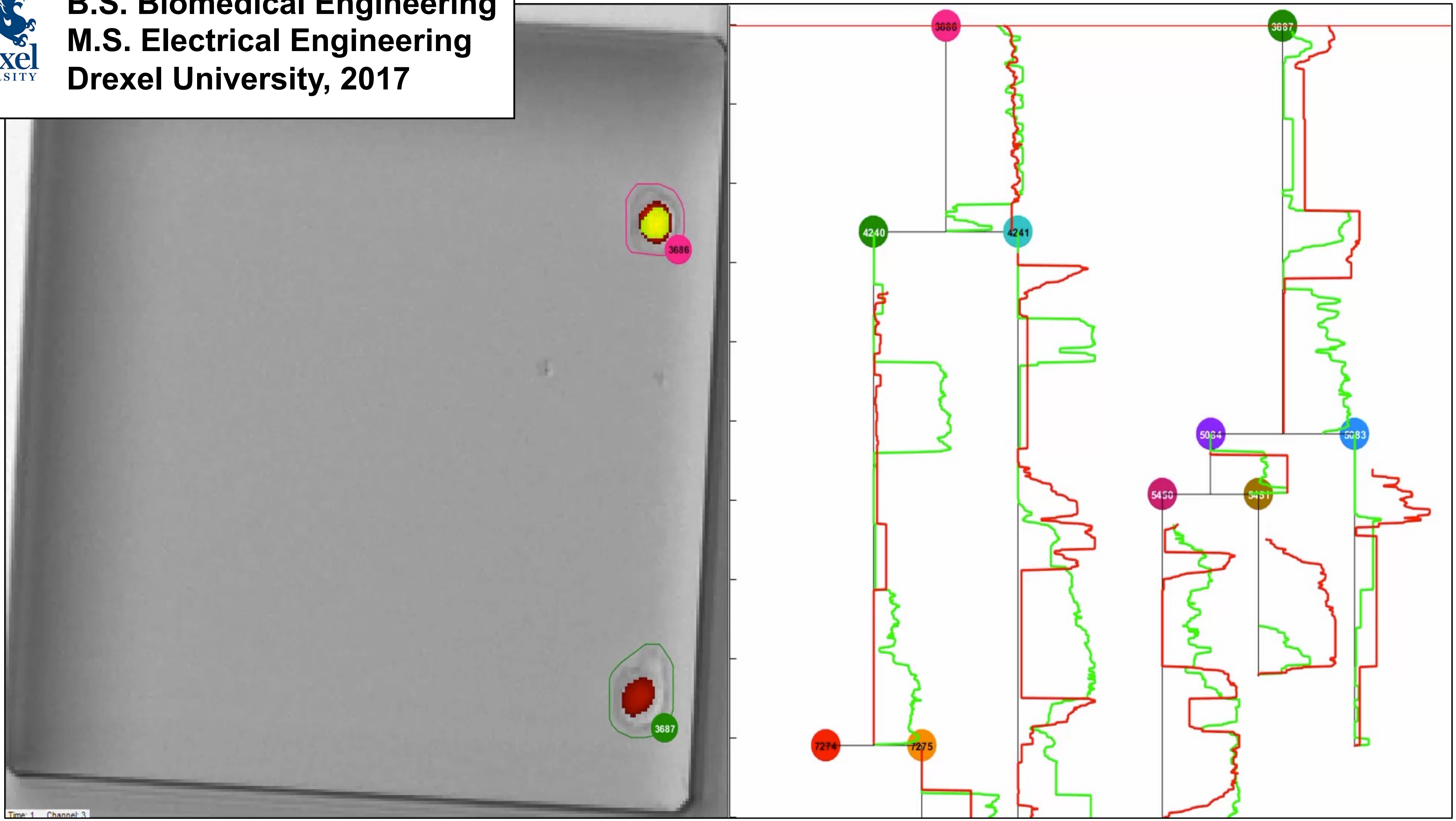
Contents

- Background
- Reflections on CT
- Related to engineering
- Applications!
- ACT outreach for engineers
- Acknowledgements

About Me

B.S. Biomedical Engineering
M.S. Electrical Engineering
Drexel University, 2017

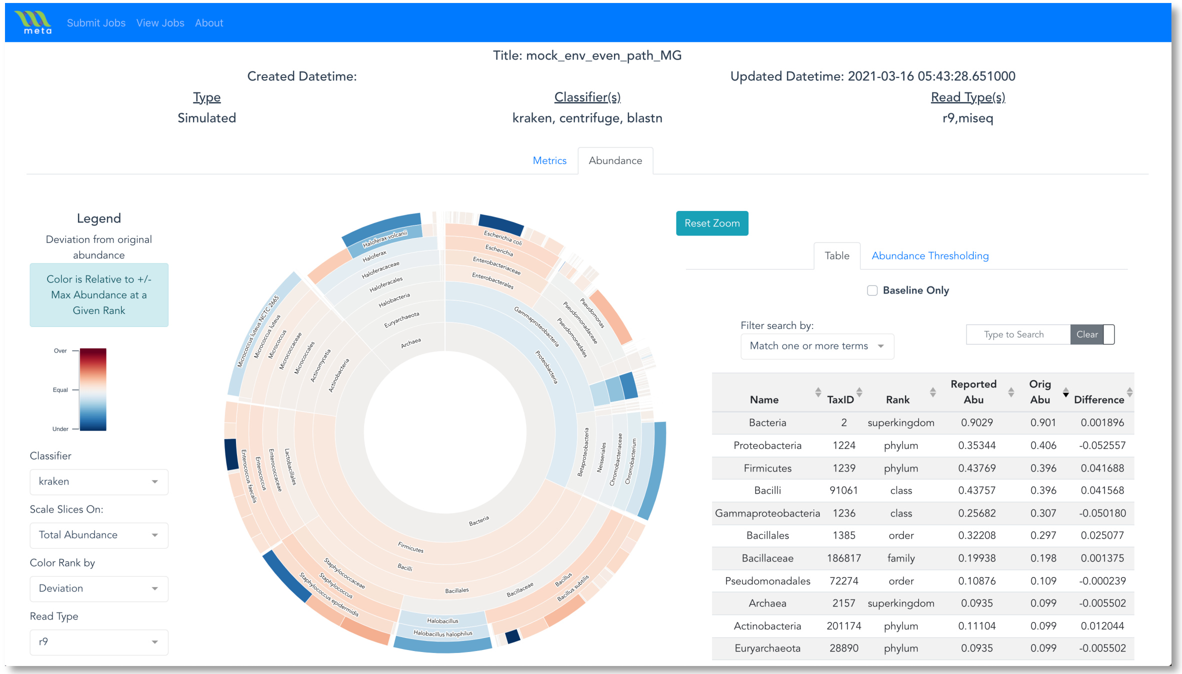
LEVER: stem cell segmentation, tracking, and lineaging. Bioimage Lab. Drexel University.



JOHNS HOPKINS
APPLIED PHYSICS LABORATORY

Software Engineer Project Lead

Humanitarian Assistance and Disaster Relief (HADR) Data Products for Hurricane Dorian 2019 Response

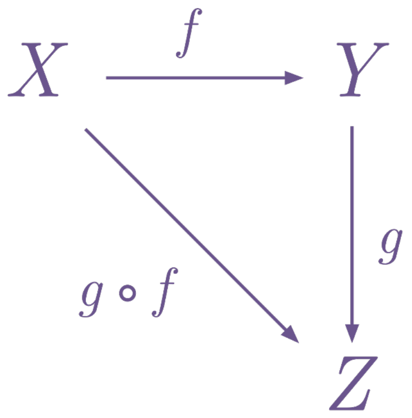


Metagenomic Evaluation, Testing, and Evaluation (META) Tool. Sunburst visualization.

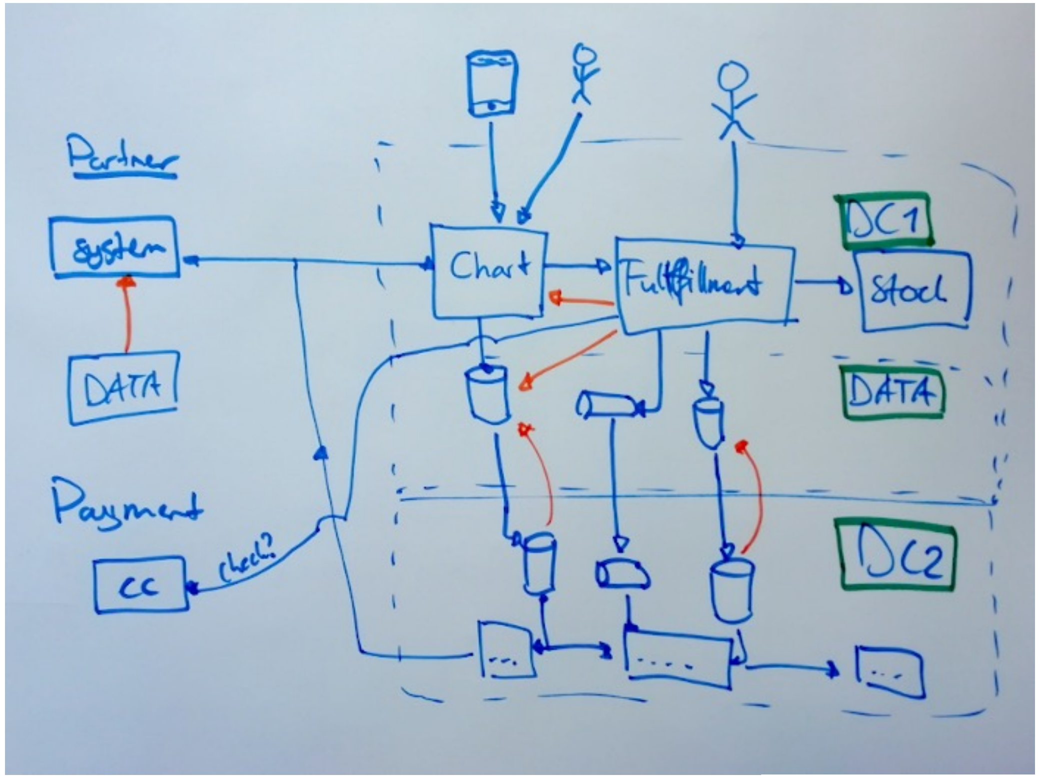
UNIVERSITY OF MARYLAND 1856

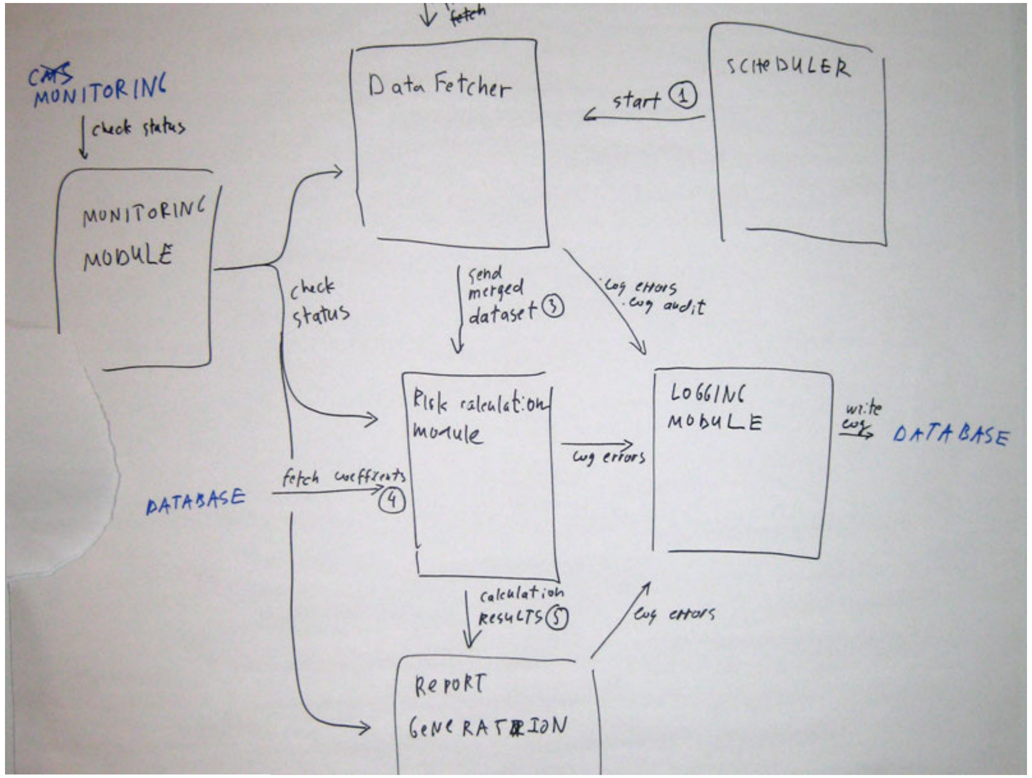
3rd Year Ph.D. Student

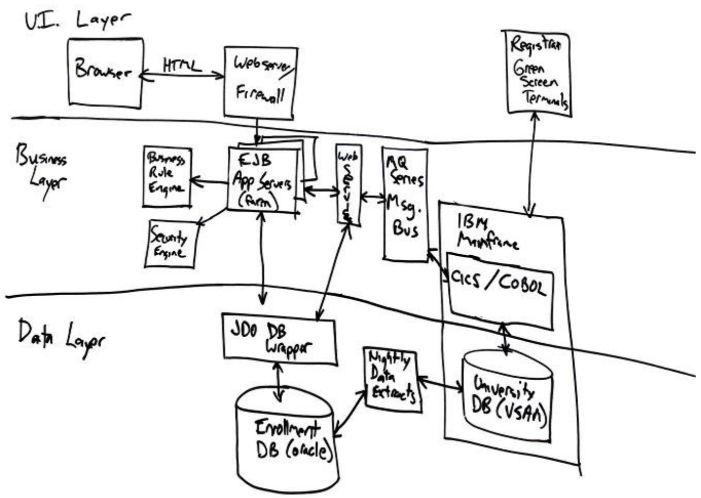
Computer Science



How did I get here?

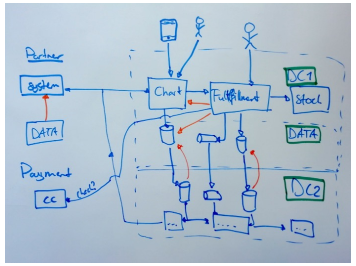

Partner
system
DATA
Payment
cc
check?
Chart
Fulfillment
Stock
DC1
DATA
DC2

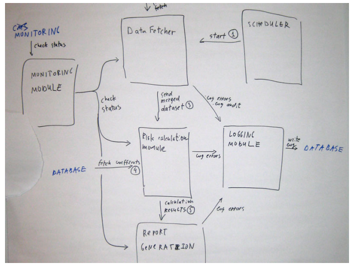

MONITORING
check status
MONITORING MODULE
Data Fetcher
SCHEDULER
start ①
check status
send merged dataset ③
log errors
log audit
Risk calculation module
LOGGING MODULE
write log
DATABASE
log errors
DATABASE
fetch coefficients ④
calculation results ⑤
log errors
REPORT GENERATION

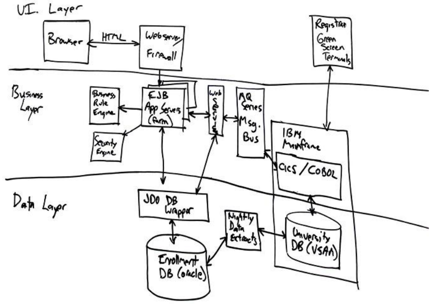

UI Layer
Browser
HTML
Webserver Firewall
Registrar Green Screen Terminals
Business Layer
Business Rule Engine
EJB App Servers (Farm)
MQ Series Msg. Bus
Security Engine
IBM Mainframe
CICS / COBOL
Data Layer
JDO DB Wrapper
Nightly Data Extracts
Enrollment DB (oracle)
University DB (VSAM)

Generalization

Mathematics







Engineering

Implementation

So, how has it been?

nLab

***Proposition 2.3*. (Yoneda lemma)**

Let $\mathscr{C}$ be a locally small category, with category of presheaves denoted $[\mathscr{C}^{\mathrm{op}}, \mathrm{Set}]$, according to Def. 2.1.

For $X \in [\mathscr{C}^{\mathrm{op}}, \mathrm{Set}]$ any presheaf, there is a canonical isomorphism

$$\mathrm{Hom}_{[C^{\mathrm{op}},\mathrm{Set}]}(y(c), X) \;\simeq\; X(c)$$

between the hom-set of presheaf homomorphisms from the representable presheaf $y(c)$ to X, and the value of X at c.

This is the standard notation used mostly in pure category theory and enriched category theory. In other parts of the literature it is customary to denote the presheaf represented by c as h_c. In that case the above is often written

$$\mathrm{Hom}(h_c, X) \simeq X(c)$$

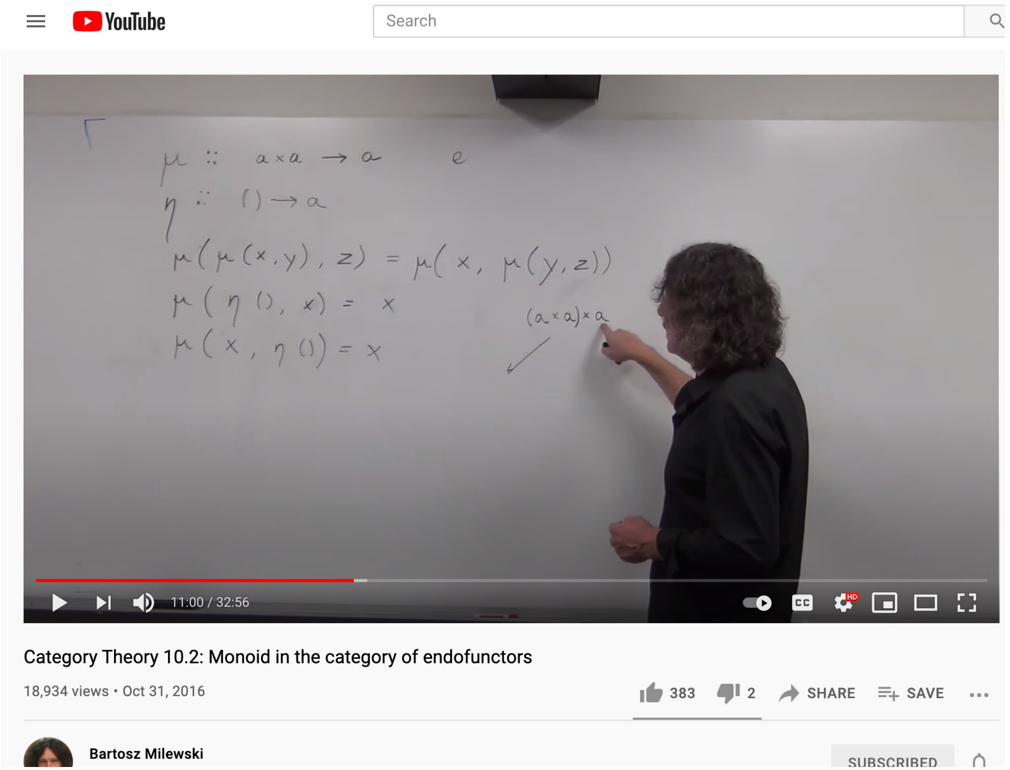
or

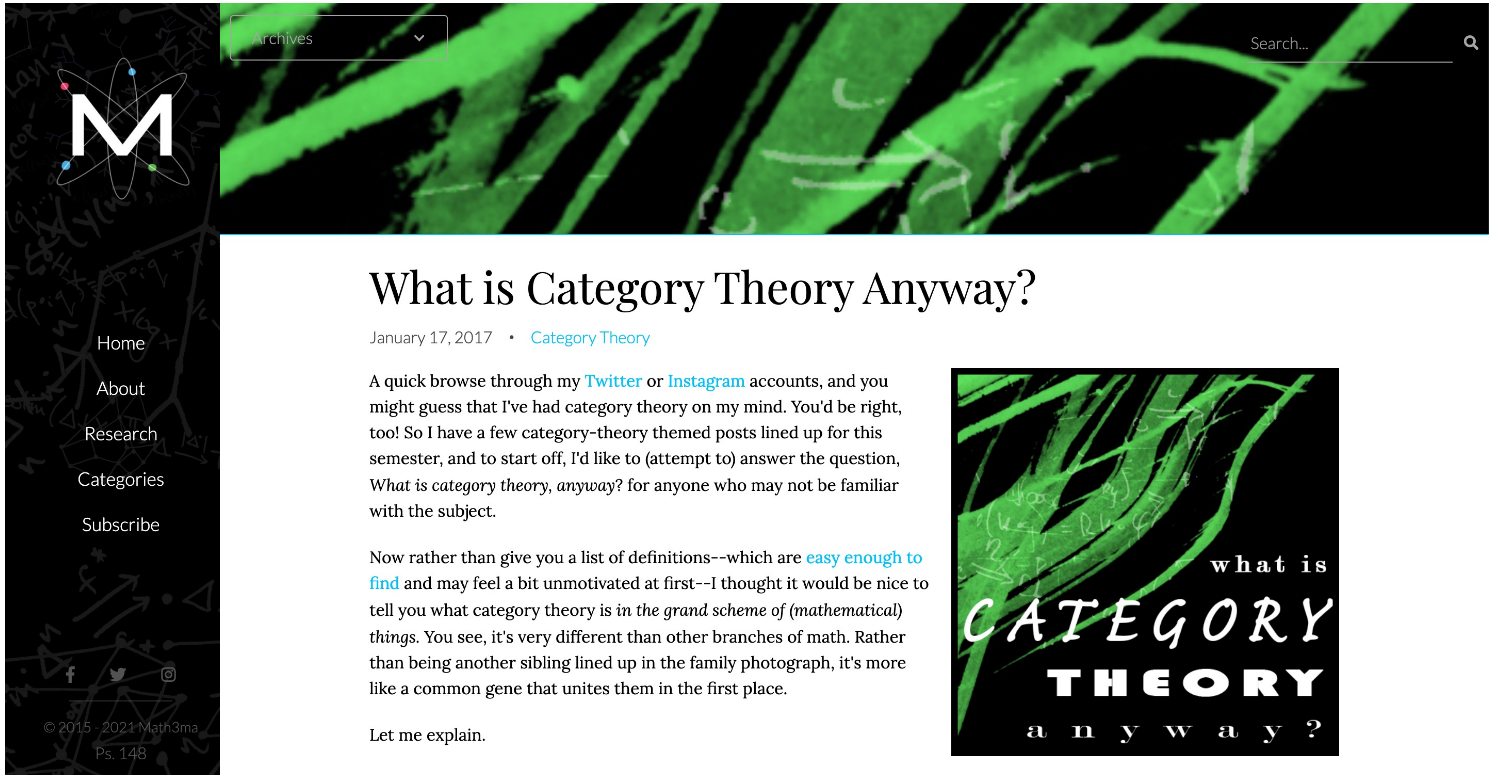
$$\mathrm{Nat}(h_c, X) \simeq X(c)$$

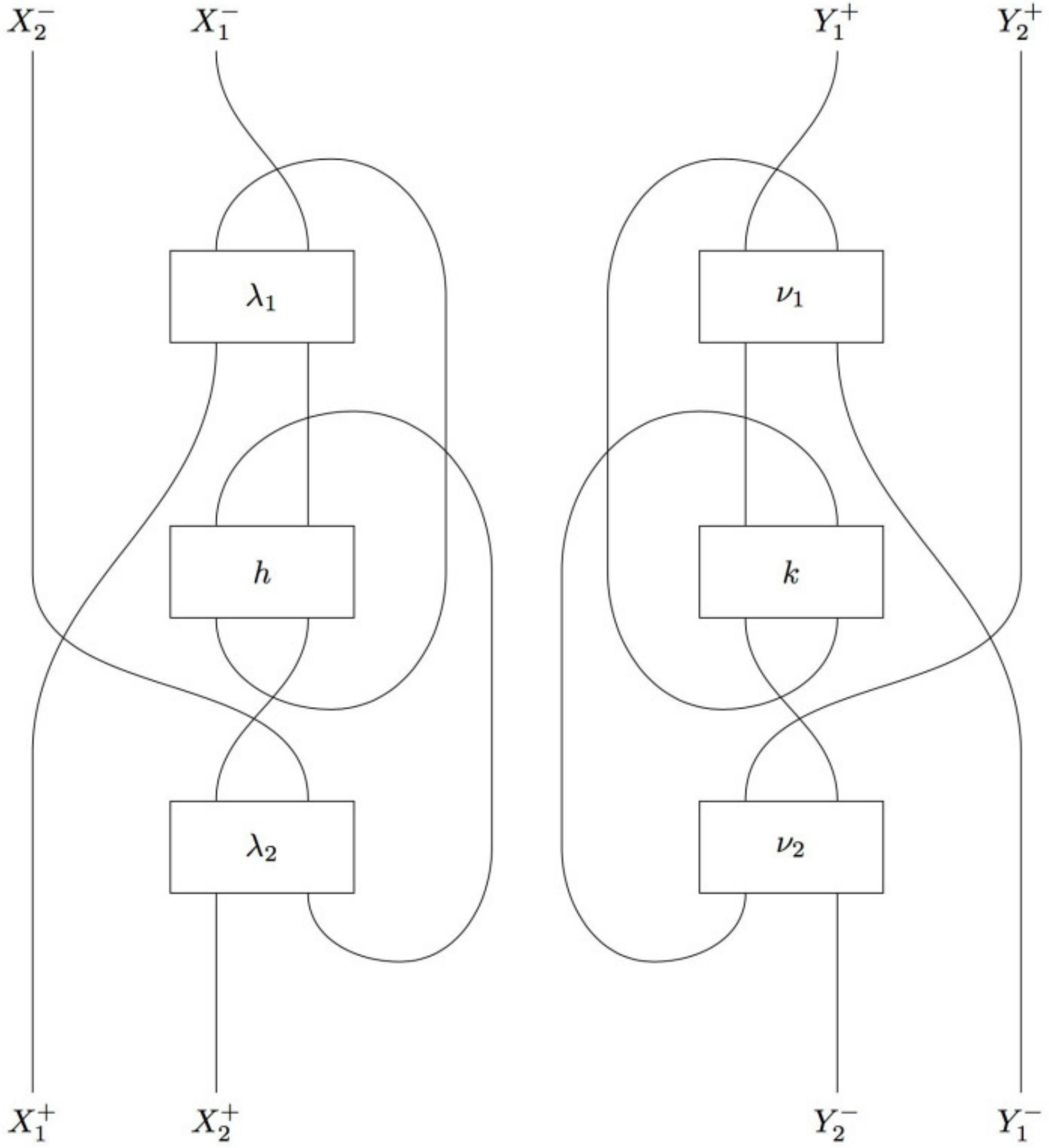
to emphasize that the morphisms of presheaves are natural transformations of the corresponding functors.

***Proof*.** The proof is by chasing the element $\mathrm{Id}_c \in C(c, c)$ around both legs of a naturality square for a natural transformation $\eta : C(-, c) \to X$ (hence a homomorphism of presheaves):

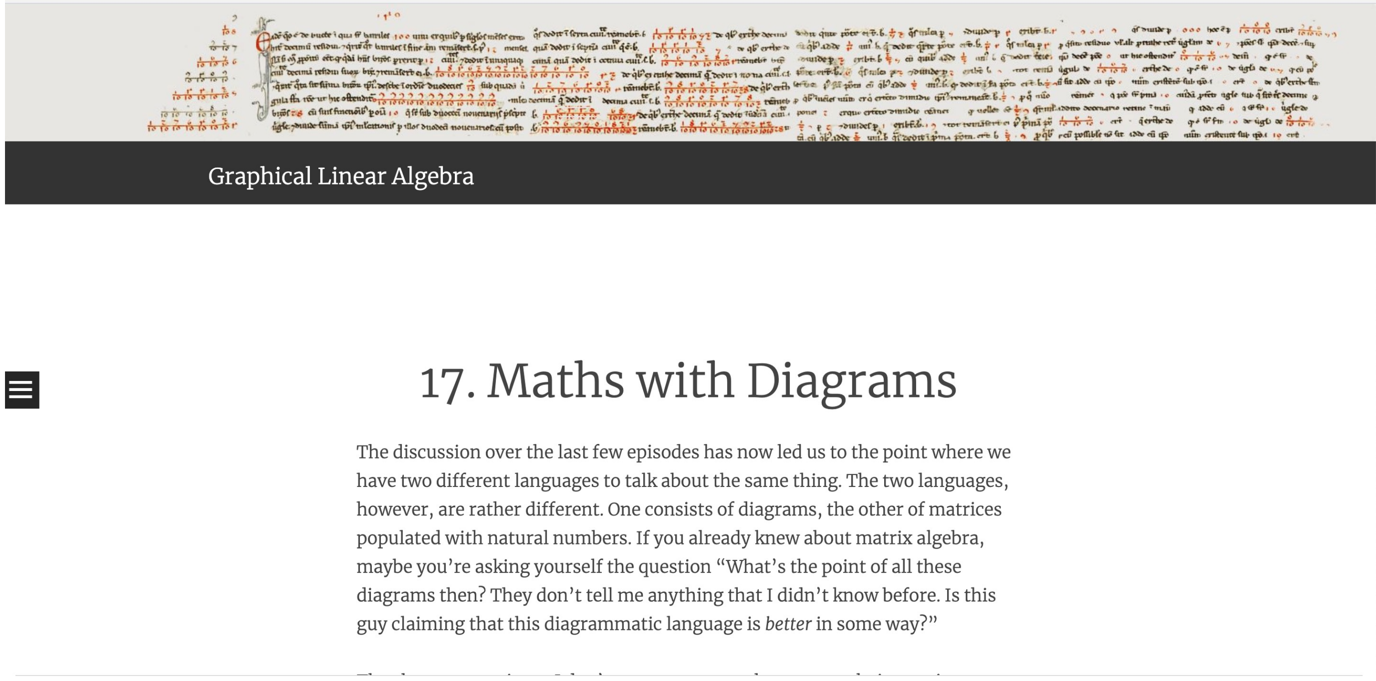
$$\begin{array}{ccccccc} C(c,c) & \overset{\eta_c}{\to} & X(c) & & \mathrm{Id}_c & \mapsto & \eta_c(\mathrm{Id}_c) \overset{\mathrm{def}}{=} \xi \\ {}^{C(f,c)}\downarrow & & \downarrow^{X(f)} & & \downarrow & & \downarrow^{X(f)} \\ C(b,c) & \underset{\eta_b}{\to} & X(b) & & f & \mapsto & \eta_b(f) \end{array}$$


YouTube
Search
11:00 / 32:56
Category Theory 10.2: Monoid in the category of endofunctors
18,934 views • Oct 31, 2016
383
2
SHARE
SAVE
Bartosz Milewski
SUBSCRIBED

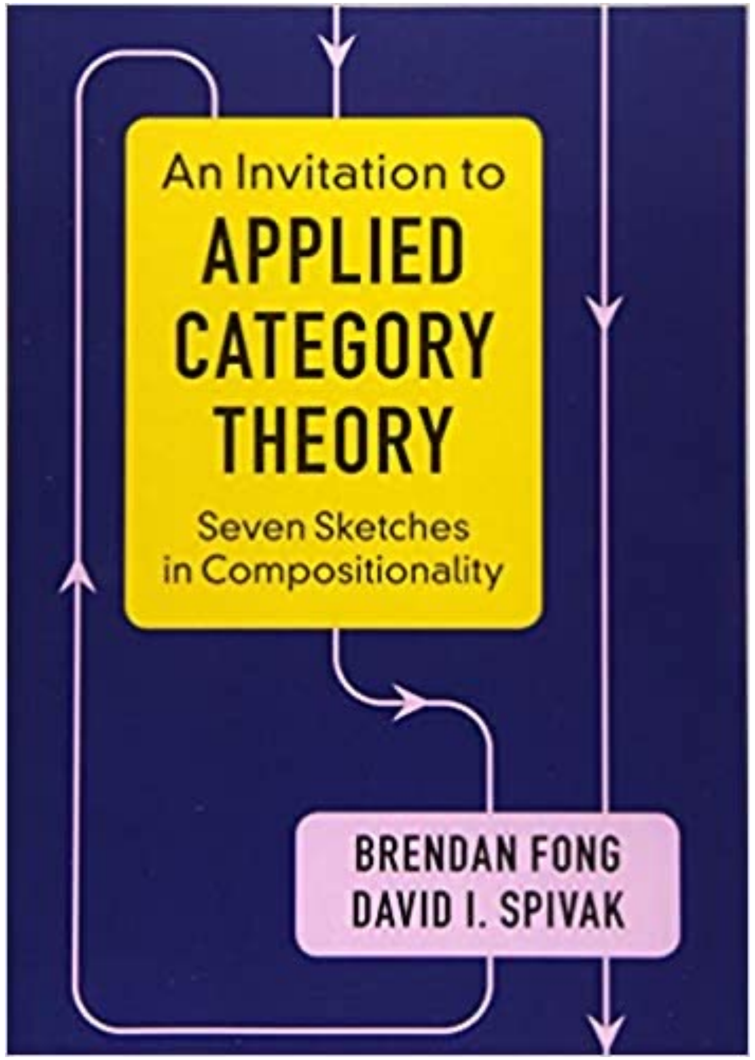

Archives
Search...
Home
About
Research
Categories
Subscribe
© 2015 - 2021 Math3ma
Ps. 148
What is Category Theory Anyway?
January 17, 2017 • Category Theory
A quick browse through my Twitter or Instagram accounts, and you might guess that I've had category theory on my mind. You'd be right, too! So I have a few category-theory themed posts lined up for this semester, and to start off, I'd like to (attempt to) answer the question, What is category theory, anyway? for anyone who may not be familiar with the subject.
Now rather than give you a list of definitions--which are easy enough to find and may feel a bit unmotivated at first--I thought it would be nice to tell you what category theory is in the grand scheme of (mathematical) things. You see, it's very different than other branches of math. Rather than being another sibling lined up in the family photograph, it's more like a common gene that unites them in the first place.
Let me explain.
what is
CATEGORY
THEORY
a n y w a y ?

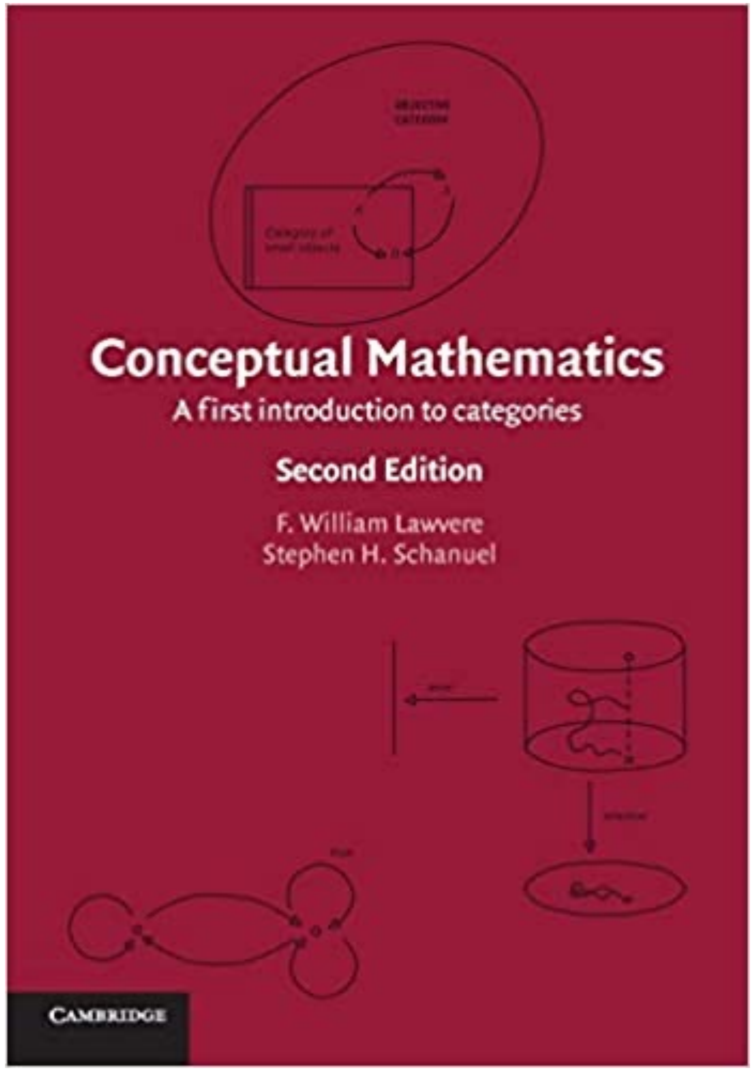


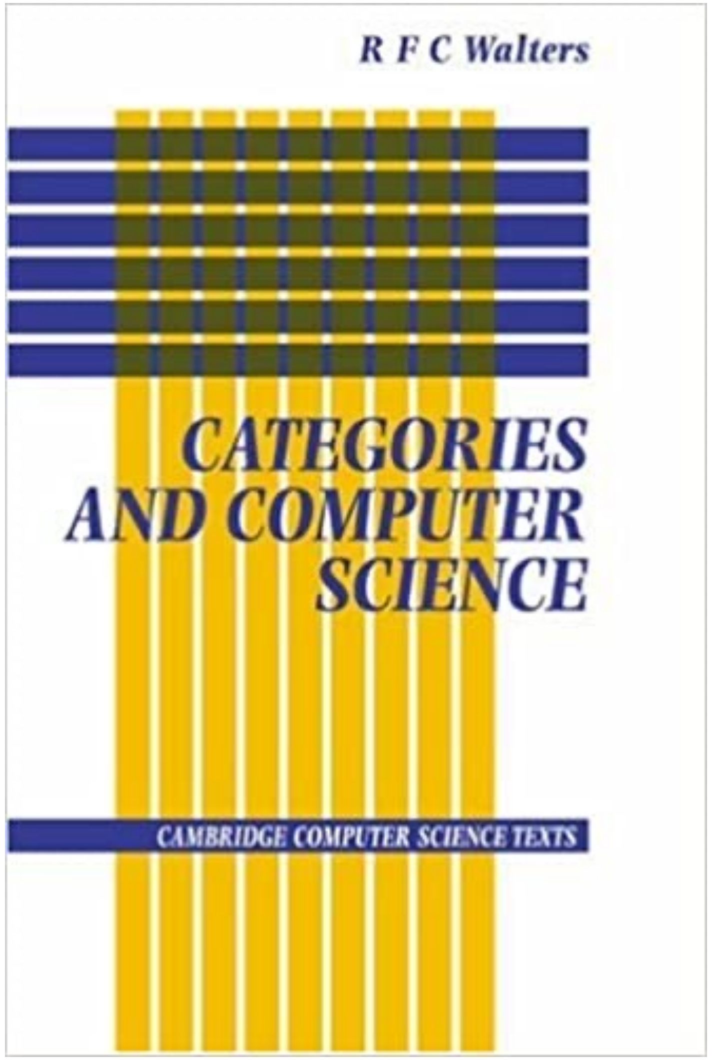
X_2^-
X_1^-
Y_1^+
Y_2^+
λ_1
ν_1
h
k
λ_2
ν_2
X_1^+
X_2^+
Y_2^-
Y_1^-

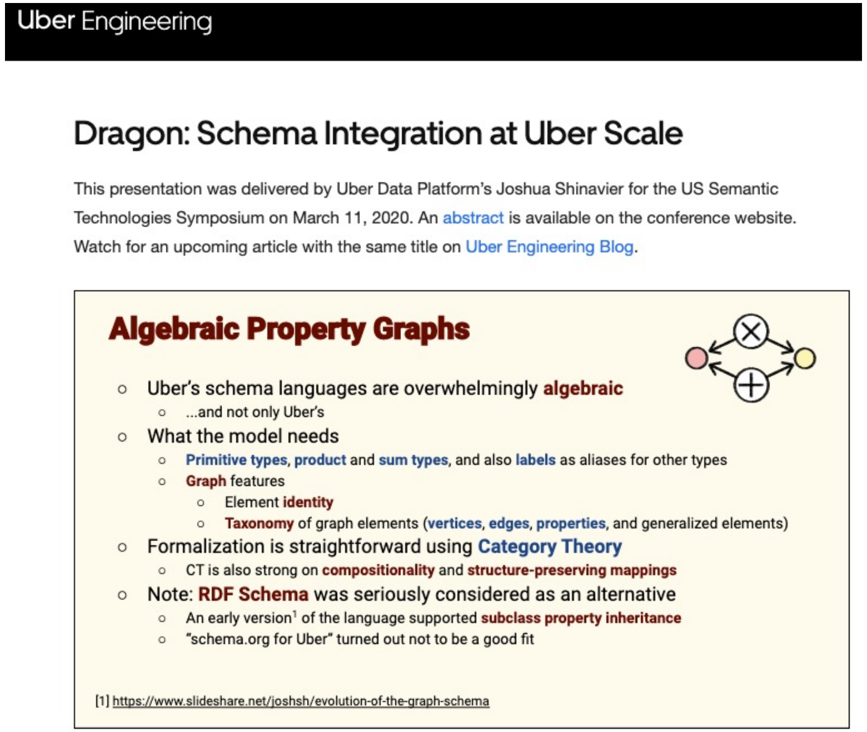


Graphical Linear Algebra
17. Maths with Diagrams
The discussion over the last few episodes has now led us to the point where we have two different languages to talk about the same thing. The two languages, however, are rather different. One consists of diagrams, the other of matrices populated with natural numbers. If you already knew about matrix algebra, maybe you're asking yourself the question "What's the point of all these diagrams then? They don't tell me anything that I didn't know before. Is this guy claiming that this diagrammatic language is *better* in some way?"


An Invitation to
APPLIED CATEGORY THEORY
Seven Sketches in Compositionality
BRENDAN FONG
DAVID I. SPIVAK


Conceptual Mathematics
A first introduction to categories
Second Edition
F. William Lawvere
Stephen H. Schanuel
CAMBRIDGE


R F C Walters
CATEGORIES AND COMPUTER SCIENCE
CAMBRIDGE COMPUTER SCIENCE TEXTS



Uber Engineering

Dragon: Schema Integration at Uber Scale

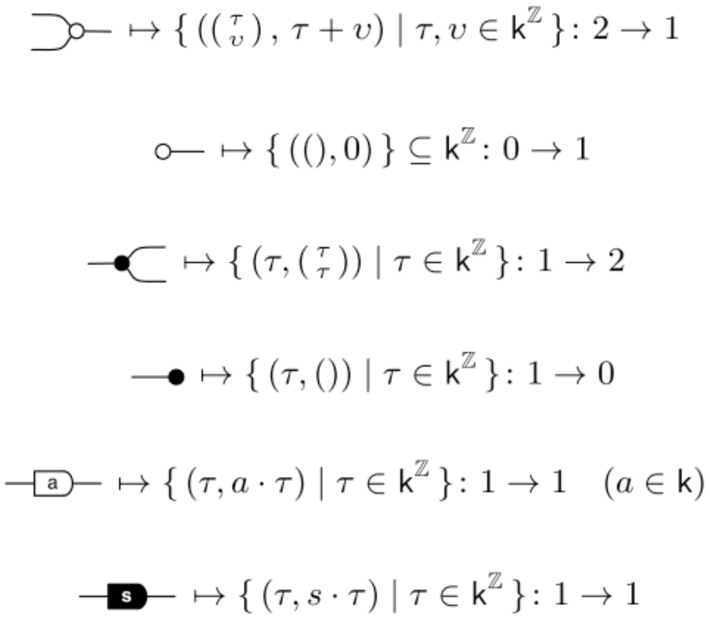
This presentation was delivered by Uber Data Platform's Joshua Shinavier for the US Semantic Technologies Symposium on March 11, 2020. An abstract is available on the conference website. Watch for an upcoming article with the same title on Uber Engineering Blog.

Algebraic Property Graphs

- Uber's schema languages are overwhelmingly **algebraic**
  - ...and not only Uber's
- What the model needs
  - **Primitive types**, **product** and **sum types**, and also **labels** as aliases for other types
  - **Graph** features
    - Element **identity**
    - **Taxonomy** of graph elements (**vertices**, **edges**, **properties**, and generalized elements)
- Formalization is straightforward using **Category Theory**
  - CT is also strong on **compositionality** and **structure-preserving mappings**
- Note: **RDF Schema** was seriously considered as an alternative
  - An early version[1] of the language supported **subclass property inheritance**
  - "schema.org for Uber" turned out not to be a good fit

[1] https://www.slideshare.net/joshsh/evolution-of-the-graph-schema

Shinavier, J., & Wisnesky, R. (2019). *Algebraic property graphs*. *arXiv* (Vol. 1). Association for Computing Machinery.



$$\mapsto \{ ((\tau, \upsilon), \tau + \upsilon) \mid \tau, \upsilon \in \mathsf{k}^{\mathbb{Z}} \} : 2 \to 1$$

$$\mapsto \{ ((), 0) \} \subseteq \mathsf{k}^{\mathbb{Z}} : 0 \to 1$$

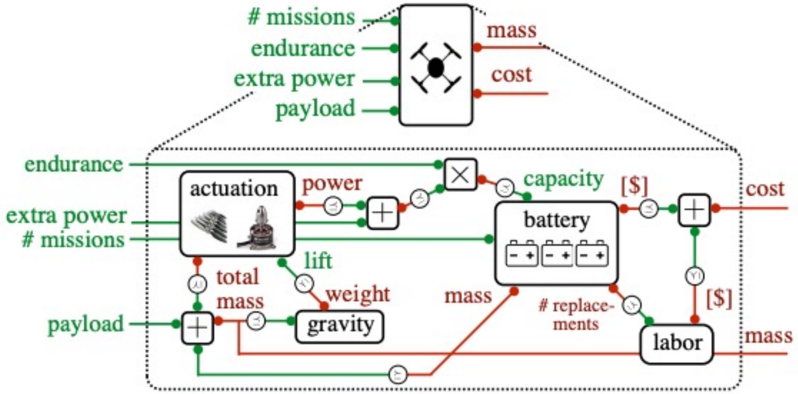
$$\mapsto \{ (\tau, (\tau, \tau)) \mid \tau \in \mathsf{k}^{\mathbb{Z}} \} : 1 \to 2$$

$$\mapsto \{ (\tau, ()) \mid \tau \in \mathsf{k}^{\mathbb{Z}} \} : 1 \to 0$$

$$\mapsto \{ (\tau, a \cdot \tau) \mid \tau \in \mathsf{k}^{\mathbb{Z}} \} : 1 \to 1 \quad (a \in \mathsf{k})$$

$$\mapsto \{ (\tau, s \cdot \tau) \mid \tau \in \mathsf{k}^{\mathbb{Z}} \} : 1 \to 1$$

Fong, B., Sobociski, P., & Rapisarda, P. (2016). A categorical approach to open and interconnected dynamical systems. *Proceedings - Symposium on Logic in Computer Science*, *05-08-July*(1)



(a) Co-design diagram corresponding to (34)–(35).

```
mcdp {
  provides endurance     [s]
  provides extra_payload [kg]
  provides extra_power   [W]
  provides num_missions  [R]
  provides velocity      [m/s]

  requires total_cost [$]
  requires total_mass [g]

  battery = instance `Batteries
  actuation = instance `Actuation

  total_power = extra_power +
    power required by actuation

  missions provided by battery >= num_missions

  energy = provided endurance * total_power
  capacity provided by battery >= energy

  total_mass = (
      mass required by battery +
      actuator_mass required by actuation +
      extra_payload)

  required total_mass >= total_mass

  gravity = 9.81 m/s^2
  weight = total_mass * gravity

  lift provided by actuation >= weight
  velocity provided by actuation >= velocity

  replacements = maintenance required by battery
  cost_per_replacement = 10 $
  labor_cost = cost_per_replacement * replacements

  required total_cost >= (
    cost required by actuation +
    cost required by battery +
    labor_cost)
}
```

(b) MCDPL code for (34)–(35). The "instance" statements refer to previously defined models for batteries (Fig. 45b) and actuation (not shown).

Andrea Censi. **A mathematical theory of co-design.** Technical Report, Laboratory for Information and Decision Systems, MIT, September 2016. Submitted and conditionally accepted to IEEE Transactions on Robotics.

What do I like about it?

Benefits

- Organization
- Graphical syntax
- Parallel universes

Challenges

- Hard to intuit
- Hard to communicate without special terminology
- Hard to get it right

What have I learned?

Examples	Category Theory Results
Operad	Define new composition operator
Natural Isomorphism	Show this thing is equal to this other thing
Decorated Cospan	Show the categorical representation of an existing algorithm has benefits
String Diagram	Define new graphical syntax
Natural Transformation /Functors	Show this thing is analogous to this other thing

How does it relate to engineering?

Engineering Problems

Security and Safety (Verification)
“Did I build the product right?”

Effectiveness (Validation)
”Did I build the the right product?”

Efficiency
“Can I make this run faster or with fewer computations?”

Comprehensibility
“Can I understand what this product is doing?”

Innovation
“How can I push the boundaries of this tools in this space?”

Examples	Category Theory Results		Engineering Problems
Operad	Define new composition operator	....	Security and Safety (Verification) *“Did I build the product right?”*
Natural Isomorphism	Show this thing is equal to this other thing	....	Effectiveness (Validation) *”Did I build the the right product?”*
Decorated Cospan	Show the categorical representation of an existing algorithm has benefits	....	Efficiency *“Can I make this run faster or with fewer computations?”*
String Diagram	Define new graphical syntax	....	Comprehensibility *“Can I understand what this product is doing?”*
Natural Transformation /Functors	Show this thing is analogous to this other thing	....	Innovation *“How can I push the boundaries of this tools in this space?”*

How can I apply category theory?

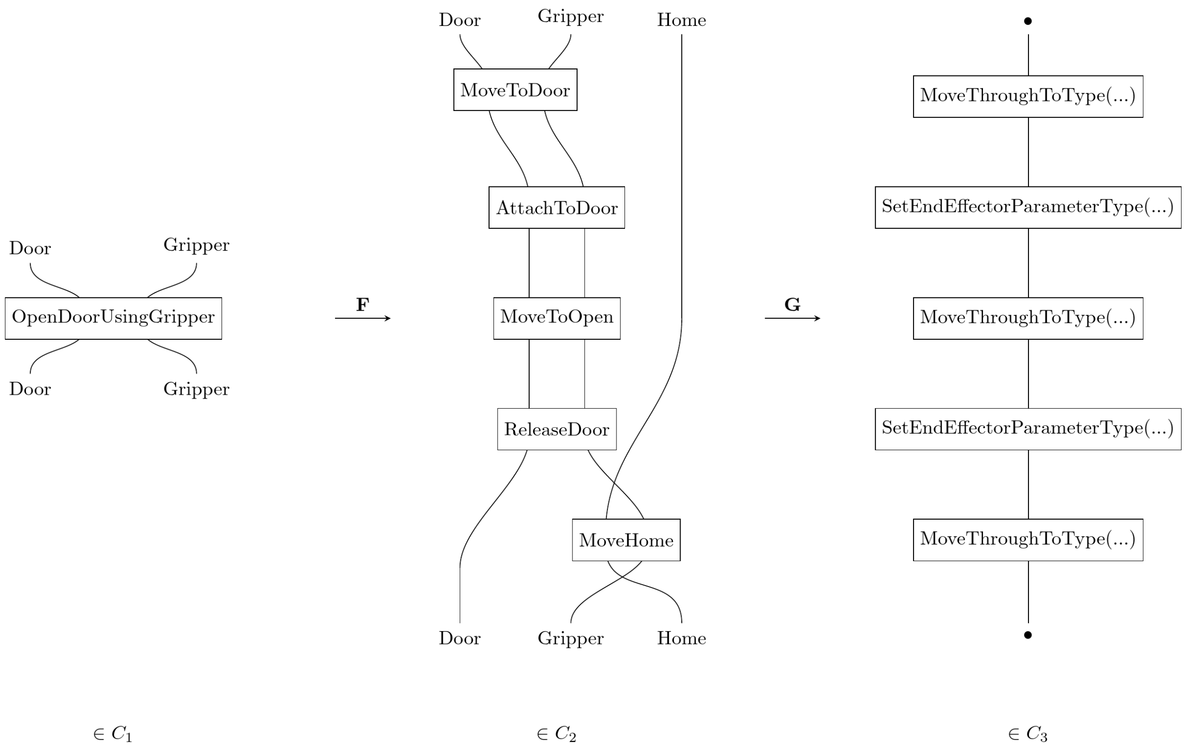
Examples	Category Theory Results		Engineering Problems	Applications
Operad	Define new composition operator	…………………	Security and Safety (Verification) *“Did I build the product right?”*	Goal-Oriented Robot Programming
Functor	Show this thing is equal to this other thing	…………………	Effectiveness (Validation) *”Did I build the the right product?”*	Modeling P-Code for Reverse Engineering
Decorated Cospan	Show there exists another map between these things	…………………	Efficiency *“Can I make this run faster or with fewer computations?”*	Analyzing Zeek Connection Log for Network Security
String Diagram	Define new graphical syntax	…………………	Comprehensibility *“Can I understand what this product is doing?”*	Visualize Classical AI Planning Solutions
Natural Transformation	Show this thing is analogous to this other thing	…………………	Innovation *“How can I push the boundaries of this tools in this space?”*	Robot Programming vs. Assembly Compiler

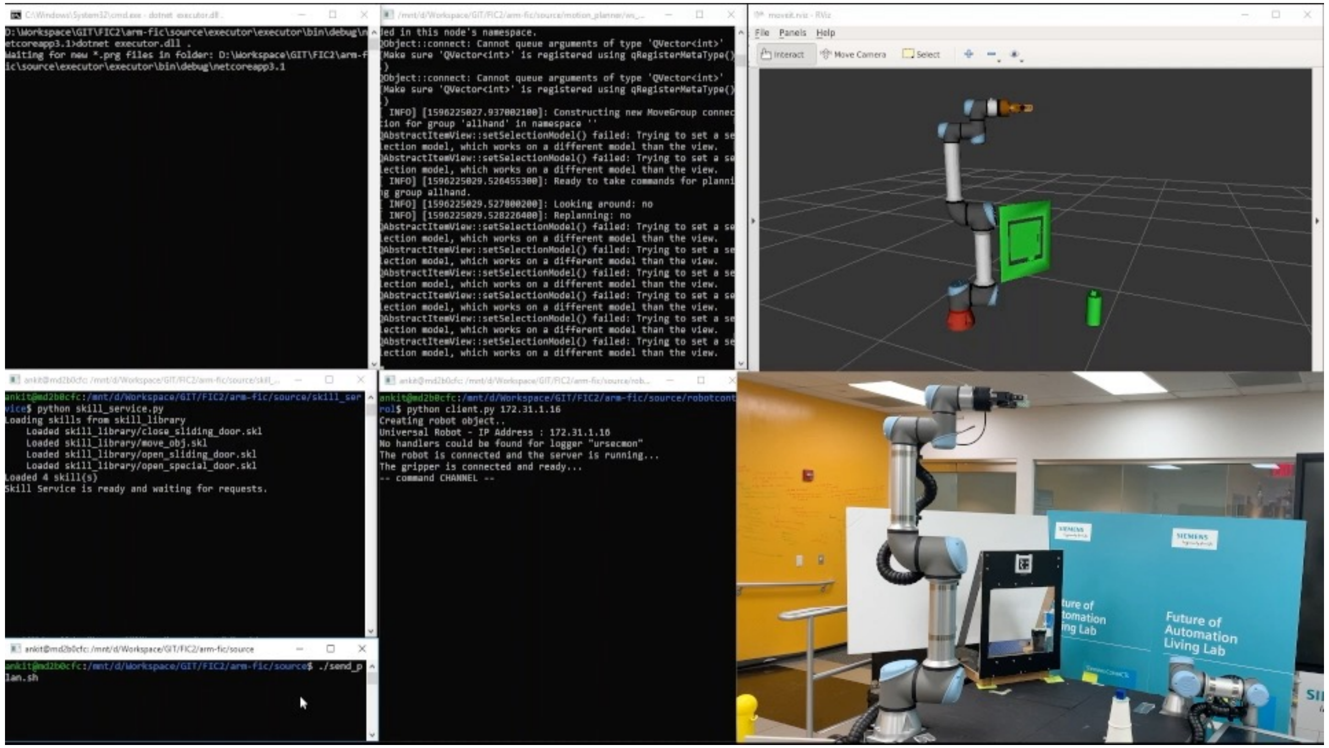
Steps for Applying Category Theory

1. Decide the CT tools
2. Align the application concepts
3. Encode the model

Goal-Oriented Robot Programming

Tools: Symmetric Monoidal Categories, Monoids, Functor





A. Aguinaldo, J. Bunker, B. Pollard, A. Canedo, G. Quiros, W. Regli. *RoboCat: A category theoretic framework for robotic interoperability using goal-oriented programming.* IEEE Transactions for Automated Science and Engineering. 2021.

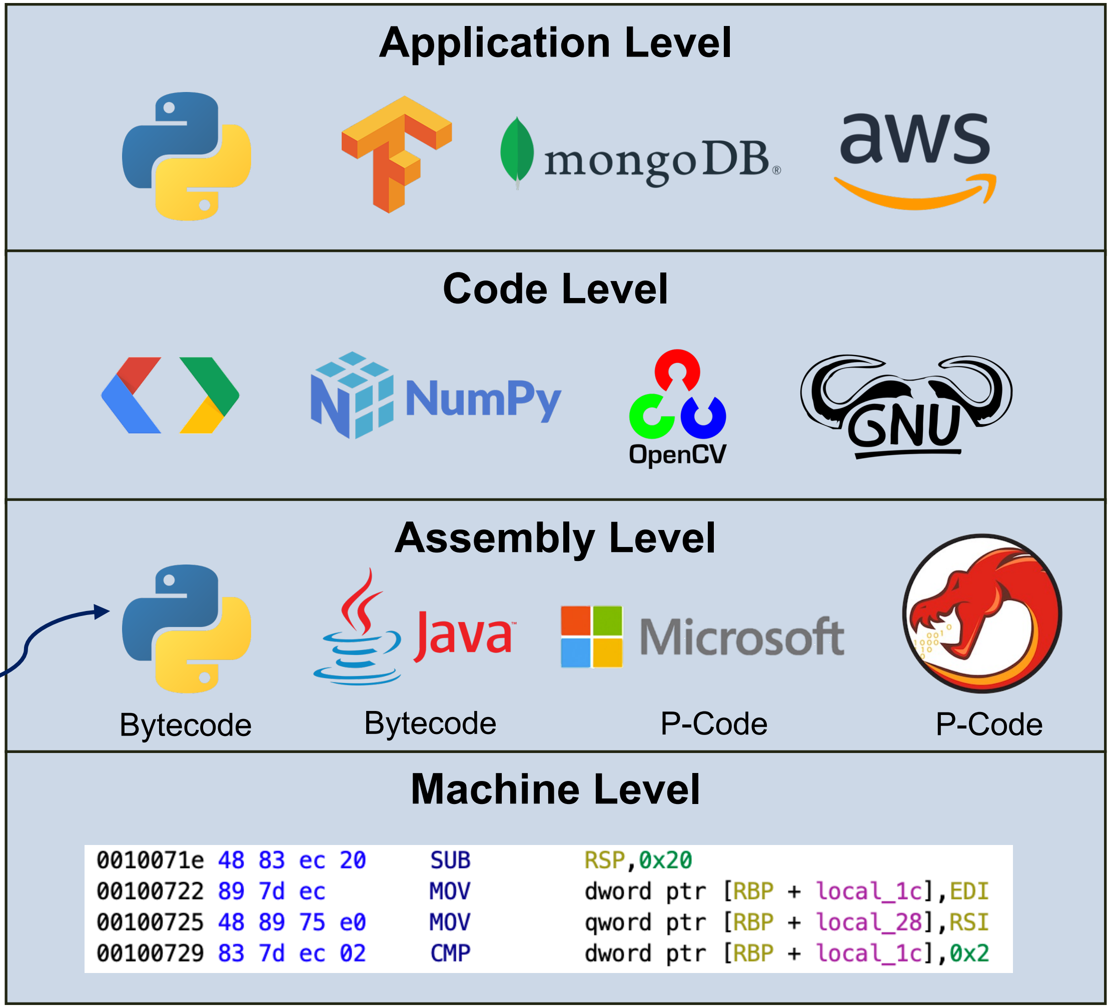
Modeling P-Code for Reverse Engineering

Tools: *(In Progress)*

States (S)
↓
Address Space Values (V)
↓
Memory Registers (R) ⟶ Virtual Memory/Pointers (P)

(P ⟶ V)

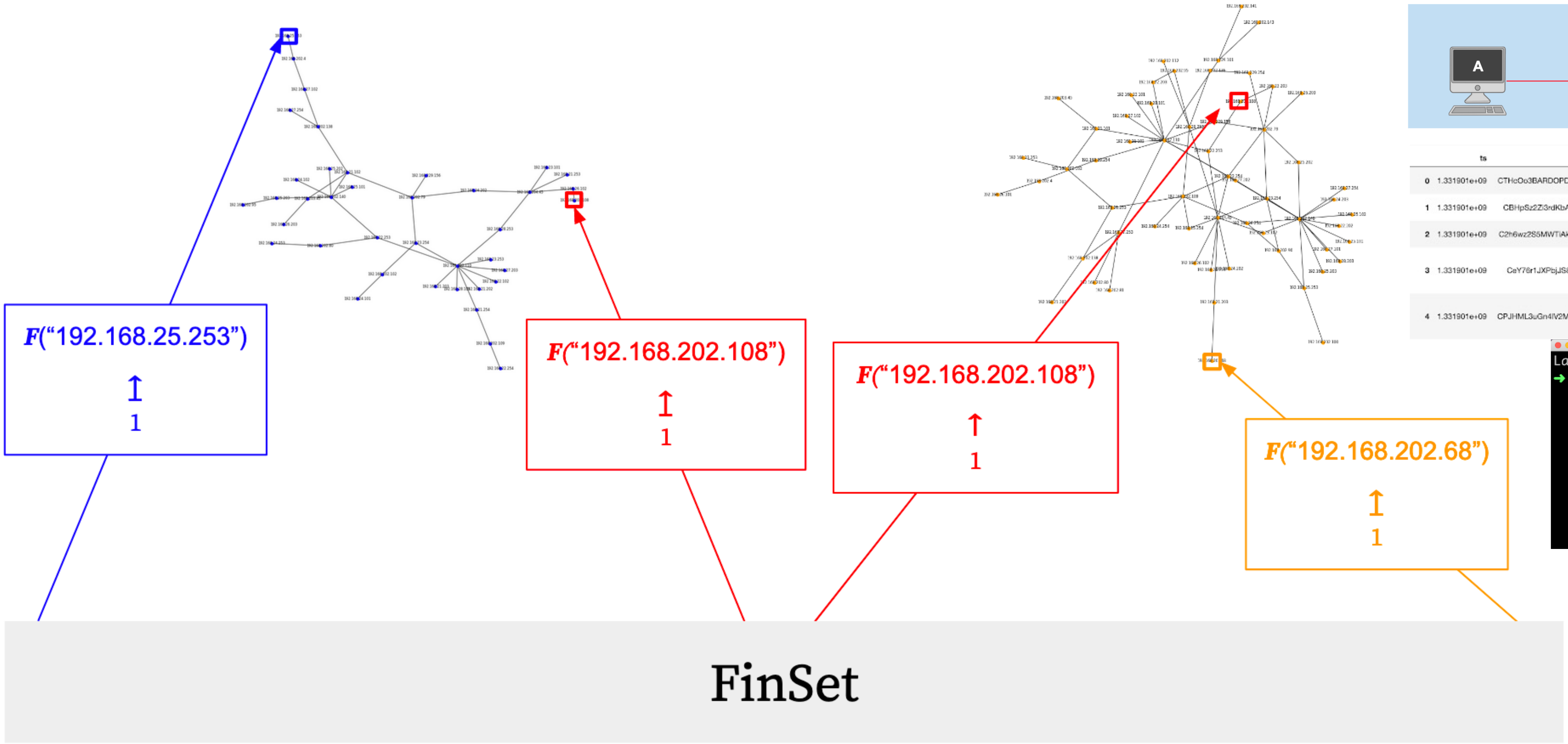
```
(unique, 0x10000110, 8) INT_ADD (register, 0x110, 8) , (const, 0x28, 8)
(unique, 0x1ff0, 8) LOAD (const, 0x1b1, 4) , (unique, 0x9e0, 8)
(unique, 0x9e0, 8) CAST (unique, 0x10000110, 8)
(unique, 0x2250, 1) INT_SLESS (register, 0x38, 4) , (const, 0x2, 4)
 ---  CBRANCH (ram, 0x100743, 1) , (unique, 0x2250, 1)
(unique, 0x1ff0, 8) LOAD (const, 0x1b1, 4) , (register, 0x30, 8)
 ---  CALL (ram, 0x1005d0, 8) , (unique, 0x100000a8, 8) , (unique, 0x1ff0, 8)
(register, 0x110, 8) INDIRECT (register, 0x110, 8) , (const, 0x32, 4)
(stack, 0xffffffffffffffc7, 4) INDIRECT (stack, 0xffffffffffffffc7, 4) , (const, 0x32, 4)
(stack, 0xffffffffffffffcb, 1) INDIRECT (stack, 0xffffffffffffffcb, 1) , (const, 0x32, 4)
(stack, 0xffffffffffffffcc, 8) INDIRECT (stack, 0xffffffffffffffcc, 8) , (const, 0x32, 4)
(stack, 0xffffffffffffffd4, 8) INDIRECT (stack, 0xffffffffffffffd4, 8) , (const, 0x32, 4)
(stack, 0xffffffffffffffdc, 8) INDIRECT (stack, 0xffffffffffffffdc, 8) , (const, 0x32, 4)
(stack, 0xffffffffffffffe4, 2) INDIRECT (stack, 0xffffffffffffffe4, 2) , (const, 0x32, 4)
(stack, 0xfffffffffffffff0, 8) INDIRECT (unique, 0x1ff0, 8) , (const, 0x32, 4)
(unique, 0x100000a8, 8) COPY (const, 0x1008a4, 8)
 ---  BRANCH (ram, 0x1007ff, 1)
```

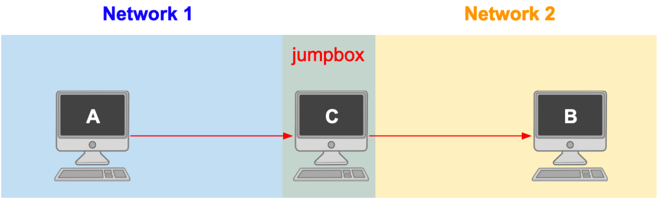


Collaboration with: G. Bakirtzis, S. Breiner, E. Subrahmanian

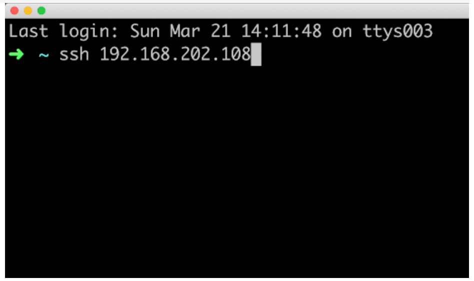
Analyzing Zeek Connection Log for Network Security

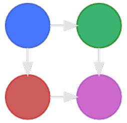
Tools: Decorated Cospans





	ts	uid	id.org_h	id.orig_p	id.resp_h	id.resp_p	status	direction
0	1.331901e+09	CTHcOo3BARDOPDjYue	192.168.202.68	53633	192.168.28.254	22	failure	INBOUND
1	1.331901e+09	CBHpSz2Zi3rdKbAvwd	192.168.202.68	35820	192.168.23.254	22	failure	INBOUND
2	1.331901e+09	C2h6wz2S5MWTiAk6Hb	192.168.202.68	36254	192.168.26.254	22	failure	INBOUND
3	1.331901e+09	CeY76r1JXPbjJS8yKb	192.168.202.68	37764	192.168.27.102	22	failure	INBOUND
4	1.331901e+09	CPJHML3uGn4lV2MGWi	192.168.202.68	40244	192.168.27.101	22	failure	INBOUND





Visualize Classical AI Planning Solutions

Tools: String diagrams

Domain file

```
(define (domain BLOCKS)
(:requirements :strips)
(:predicates (on ?x ?y)
(ontable ?x)
(clear ?x)
(handempty)
(holding ?x)
)

(:action pick-up
:parameters (?x)
:precondition (and (clear ?x)
(ontable ?x) (handempty))
:effect
(and (not (ontable ?x))
(not (clear ?x))
(not (handempty))
(holding ?x)))
...
```

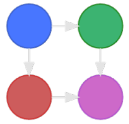
Problem file

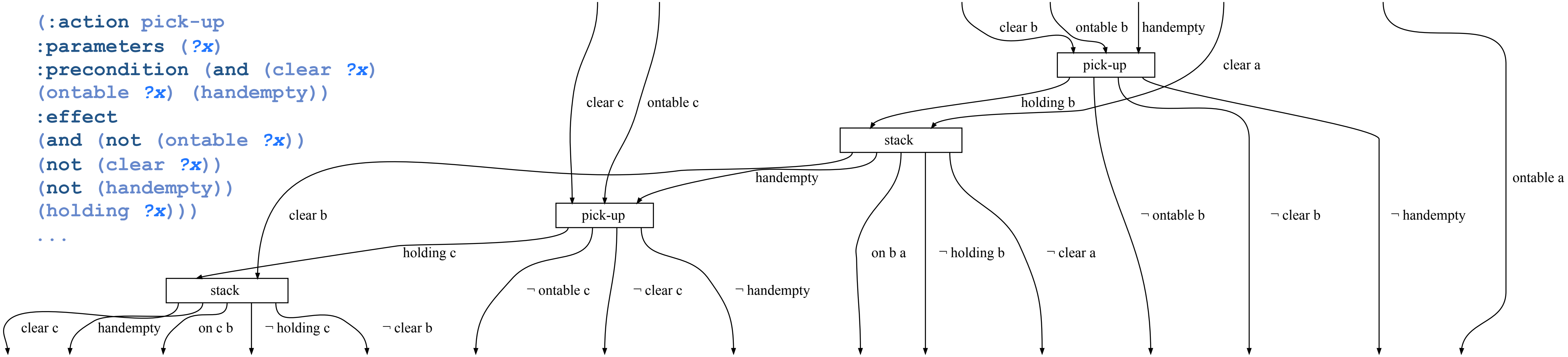
```
(define (problem BLOCKS-3-0)
(:domain BLOCKS)
(:objects a b c)
(:init (clear c) (clear a) (clear b)
(ontable c) (ontable a) (ontable b)
(handempty))
(:goal (AND (on c b) (on b a)))
)
```

Solution

```
pick-up b
stack b a
pick-up c
stack c b
```

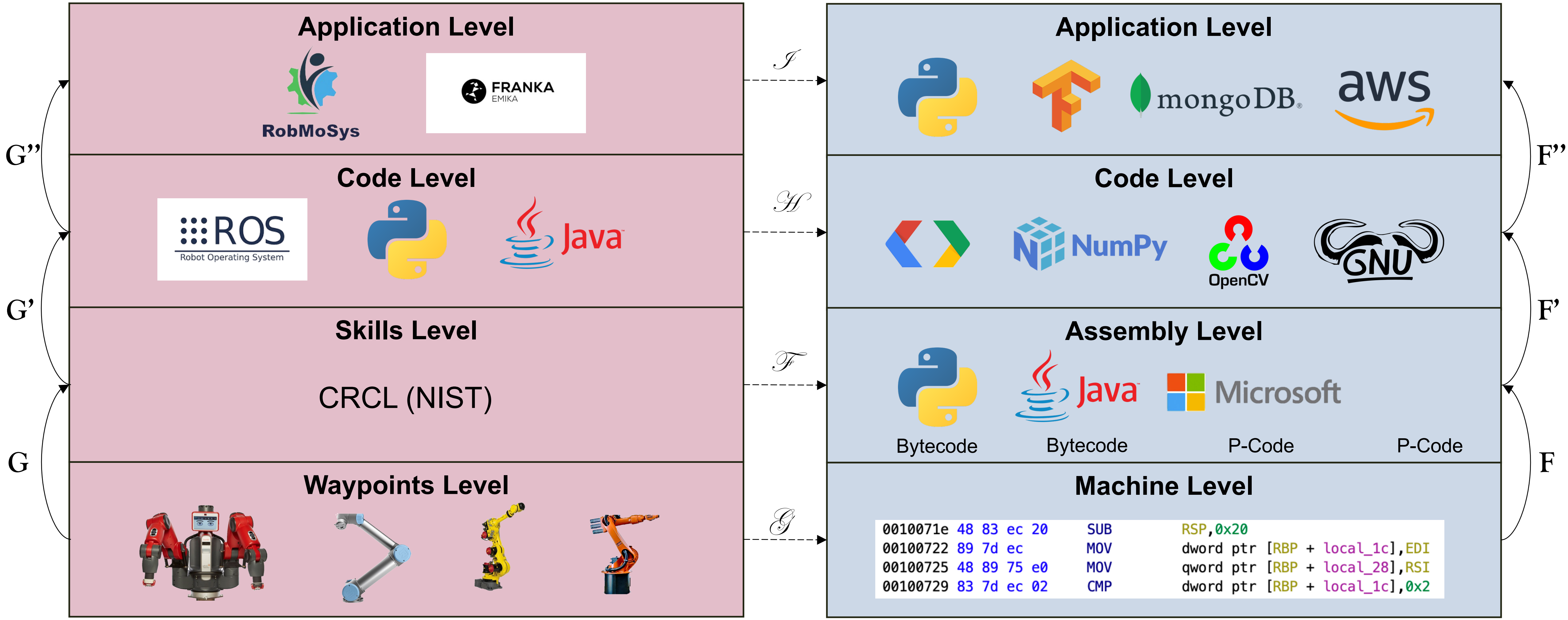
Angeline Aguinaldo and William Regli. *Encoding Compositionality in Classical Planning Solutions.* International Joint Conference for Artificial Intelligence (IJCAI) Generalization in Planning Workshop. 2021.





Robot Programming vs. Assembly Compiler

Tools: *(In Progress)*



How can we improve ACT outreach?

Describe what aspects of the problem helped you decide on the tools

Categorify existing engineering abstractions

Incorporate cognitive vs. categorical translation table

Visualize taxonomy of categories to show how they generalize or restrict

Describe what aspects of the problem helped you decide on the tools

Categorify existing engineering abstractions

Incorporate cognitive vs. categorical translation table

Visualize taxonomy of categories to show how they generalize or restrict

Describe what aspects of the problem helped you decide on the tools

Categorify existing engineering abstractions

Incorporate cognitive vs. categorical translation table

Step	Cognitive	Categorical
1	Obtain two datasets representing SSH connections for two networks, i.e. Network 1 and Network 2	Pick two objects from the category of FinSet
2	Compute two graphs using the origin IP and response IP columns of each dataset	Define functor from $\boldsymbol{F}\colon \mathrm{FinSet} \to \mathrm{Graph}$
3	Pick **jumpbox IP**	Pick span that will be the common leg
4	Pick **starting IP** in Network 1 and **target IP** in Network 2	Pick input element in cospan 1 and output element in cospan 2
5	Combine rows of datasets, but add labels saying which dataset each row came from except for the **jumpbox IP**	Compute disjoint union of finite sets quotiented by identified element images in the apexes (i.e. pushout)
6	Compute graph of combined dataset	Compute graph for pushout using decoration functor coherence maps
7	Check if **starting IP** is connected to **target IP** in graph	Use coequalizer to compute connected components

Decorated Cospans for Network Security

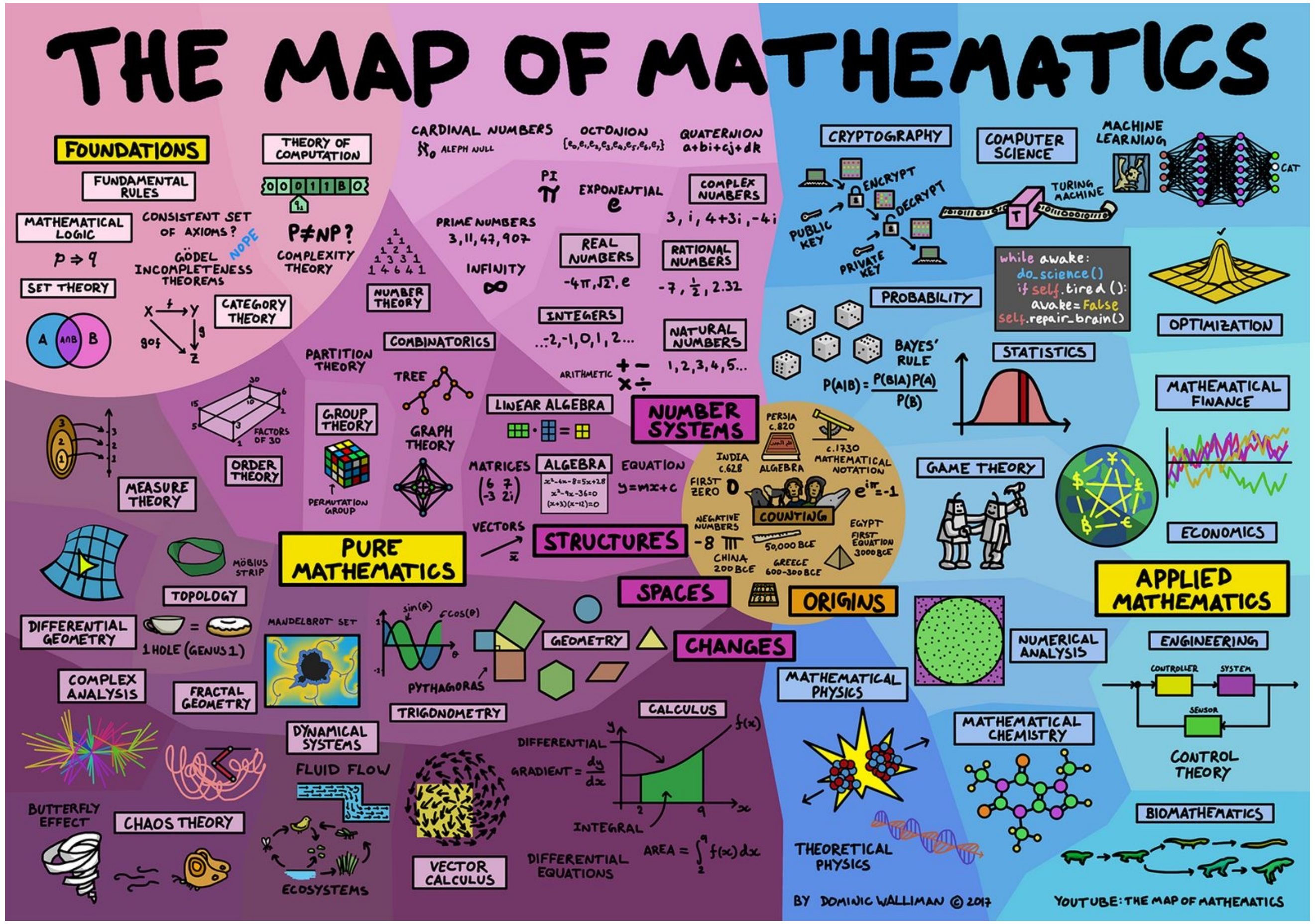
Visualize taxonomy of categories to show how they generalize or restrict

Describe what aspects of the problem helped you decide on the tools

Categorify existing engineering abstractions

Incorporate cognitive vs. categorical translation table

Visualize taxonomy of categories to show how they generalize or restrict



Domain of Science (DoS), Youtube

Acknowledgements

Spencer Breiner
Eswaran Subrahmanian
Brendan Fong
Georgios Bakirtzis
Sophie Libkind
David Jaz Meyers
Evan Patterson
James Fairbanks
Owen Lynch
Christian Williams
David Spivak
Arquimedes Canedo
Jacob Bunker
Blake Pollard
Gustavo Quiros
William Regli

Thanks for listening!

Angeline Aguinaldo
aaguinal@umd.edu
angeline.aguinaldo@jhuapl.edu