A whirlwind tour of category theory, for the working scientist and engineer

David I. Spivak

dspivak@math.mit.edu
Mathematics Department
Massachusetts Institute of Technology

Presented on 2016/11/10
at Télécom Bretagne

Outline

Outline

1. **Introduction**
   - What is category theory?
   - Analogies
   - Plan of the talk
2. Categorification
3. Something more concrete
4. Defining categories and functors
5. The whirlwind tour

What is category theory?

- Since its invention in the early 1940s, category theory (CT) has revolutionized math.
- It's a principled way of thinking about mathematical subjects.
- CT was invented to build bridges between disparate branches of math.
  - It does so by distilling the essence of mathematical structure.
  - We consider: the objects of study and the relationships between them.

What is category theory?

- Since its invention in the early 1940s, category theory (CT) has revolutionized math.
- It's a principled way of thinking about mathematical subjects.
- CT was invented to build bridges between disparate branches of math.
  - It does so by distilling the essence of mathematical structure.
  - We consider: the objects of study and the relationships between them.
- Original use: connect two disparate subjects, topology and algebra.
  - The essence of each subject was formulated as a category.
  - Rigorous relationships (functors) were established between them.
  - Functors allowed the import of theorems from algebra into topology, as *new theorems*.

Analogies

- Category theory interconnects mathematics.
  - Whole disciplines, subdisciplines, or simple thoughts can be categories.
  - Functors connect categories, building bridges between worlds.
- Distinct towns and bridges; not sprawl.

Analogies

- Category theory interconnects mathematics.
  - Whole disciplines, subdisciplines, or simple thoughts can be categories.
  - Functors connect categories, building bridges between worlds.
- Distinct towns and bridges; not sprawl.
  - Different disciplines need to develop their own language and tools.
  - We don't need everything to be put into one big pile.
  - But connections let us leverage the insights from multiple domains.
- Functors have been described as formal analogies.
  - Often two fields share some common substructures.
  - Functors make this notion of commonality precise.
  - Connectable disciplines may be neighboring, distant, or fractal.

Analogies

- Category theory interconnects mathematics.
  - Whole disciplines, subdisciplines, or simple thoughts can be categories.
  - Functors connect categories, building bridges between worlds.
- Distinct towns and bridges; not sprawl.
  - Different disciplines need to develop their own language and tools.
  - We don't need everything to be put into one big pile.
  - But connections let us leverage the insights from multiple domains.
- Functors have been described as formal analogies.
  - Often two fields share some common substructures.
  - Functors make this notion of commonality precise.
  - Connectable disciplines may be neighboring, distant, or fractal.
- Connecting disciplines is very important for solving hard problems.
- In that spirit, let's make some analogies.

Relativity and quantum mechanics

- Category theory has something in common with relativity.
  - Set theory is akin to Aristotilian space-time: one reference frame.
  - A category is a reference frame in which one sees some part of math.
  - Category theory is a multiplicity of interconnected perspectives.

Relativity and quantum mechanics

- Category theory has something in common with relativity.
  - Set theory is akin to Aristotilian space-time: one reference frame.
  - A category is a reference frame in which one sees some part of math.
  - Category theory is a multiplicity of interconnected perspectives.
- Category theory has something in common with quantum mechanics (QM).
  - What might a non-physicist secretly think about QM?
  - It's strange, esoteric, possibly wrong or incomplete, not relevant to me.
  - But note that QM is ubiquitous to the point of mundane:
    - Lasers at the grocery store.
    - LEDs in your flashlight.
    - Transistors in your phone.
    - You can't create any of these without a deep understanding of QM.

Relativity and quantum mechanics

- Category theory has something in common with relativity.
  - Set theory is akin to Aristotilian space-time: one reference frame.
  - A category is a reference frame in which one sees some part of math.
  - Category theory is a multiplicity of interconnected perspectives.
- Category theory has something in common with quantum mechanics (QM).
  - What might a non-physicist secretly think about QM?
  - It's strange, esoteric, possibly wrong or incomplete, not relevant to me.
  - But note that QM is ubiquitous to the point of mundane:
    - Lasers at the grocery store.
    - LEDs in your flashlight.
    - Transistors in your phone.
    - You can't create any of these without a deep understanding of QM.
- In modern mathematics, CT is ubiquitous and essential.
  - It might seem strange at first, but it's not going away.
  - And scientists and engineers are starting to realize its value.

Category theory: branching out

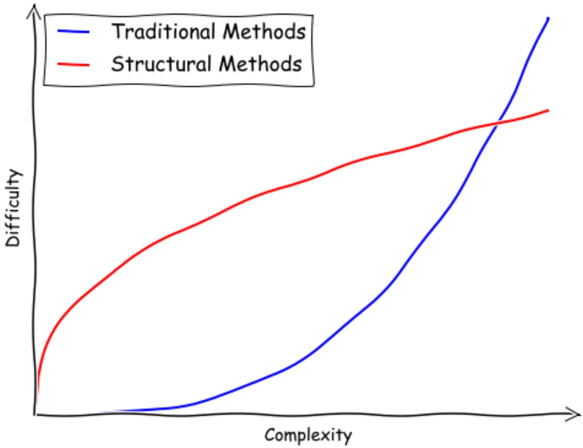
- Category theory naturally fosters connections between disparate fields.
- It has branched out of math and into:
  - physics,
  - linguistics,
  - materials science,
  - biology,
  - manufacturing,
  - economics and game theory, etc.
- CT has had most success in computer science, formalizing programming languages.

Category theory at NIST

NIST is the US national institute of standards and technology.

- They are concerned with
  - Information representation
  - Model integration
  - Multiple scales and formalisms
- They have identified category theory as a potential foundation.
- Here's how they put it, in cartoon form:

NIST: "The need for categorical engineering"



Breiner, S., Jones A., Subrahmanian, E. "Categories for the working engineer: A call for applied category theory." *International Category Theory Conference 2016*, Halifax.

Applications we'll discuss today

In the abstract to the talk I mentioned:

- hierarchies,
- symmetries,
- agent actions,
- information and communication,
- local-to-global relationships, and
- compositionality.

We'll get to each of these concepts along the way, in no particular order.

Outline

Categorification in mathematics

- Over the last 20 years, many fields of math have been *categorified*.
  - This means reformulated at a “higher categorical level”.
  - Equality is replaced by isomorphism.
  - This program has led to many new results across mathematics.
- Is $2+2$ equal to 4 or are they just isomorphic?
  - Let’s look at our hands.
  - Can the difference possibly matter?

Categorification in mathematics

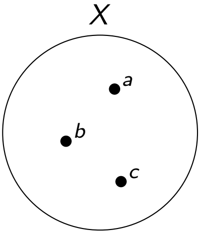
- Over the last 20 years, many fields of math have been *categorified*.
  - This means reformulated at a "higher categorical level".
  - Equality is replaced by isomorphism.
  - This program has led to many new results across mathematics.
- Is 2+2 equal to 4 or are they just isomorphic?
  - Let's look at our hands.
  - Can the difference possibly matter? The answer is yes!
- We will begin by categorifying the natural numbers:

$$\mathbb{N} = \{0, 1, 2, 3, 4, \ldots, 871^{23}, \ldots, n, \ldots\}$$

Finite sets

I assume everyone knows what a set is, but let's recall.

- A set is a bag of dots.
  - The set $X = \{a, b, c\}$ is shown to the right.
  - An element of X is denoted $x \in X$.
  - The *cardinality* of X is the number of elements, denoted $|X|$. Here $|X| = 3$.
  - The *empty set* $\emptyset$ is a bag with no dots.
- Finiteness: X is finite; $\mathbb{N} = \{0, 1, 2, \ldots\}$ is not.
- For any $n \in \mathbb{N}$, define $\underline{n} := \{1, \ldots, n\}$, finite!
  - So, for example, $\underline{0} = \emptyset$.



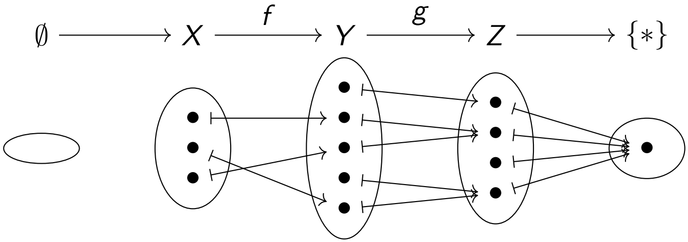
Categorification idea: what we call "numbers" are just cardinalities.

Functions

- A function from X to Y is an assignment:
  - Each $x \in X$ is assigned exactly one $y \in Y$.
  - We write $x \mapsto y$ and say x is *sent to* y.
- If the function is called f, we write $f\colon X \to Y$ or $X \xrightarrow{f} Y$,
  - and $f(x) = y$ means f sends x to y, or $f\colon x \mapsto y$.

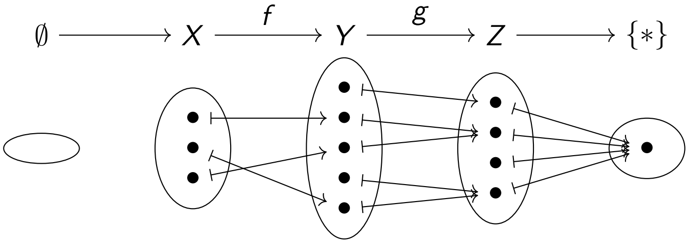
Functions

- A function from X to Y is an assignment:
  - Each $x \in X$ is assigned exactly one $y \in Y$.
  - We write $x \mapsto y$ and say x is *sent to* y.
- If the function is called f, we write $f\colon X \to Y$ or $X \xrightarrow{f} Y$,
  - and $f(x) = y$ means f sends x to y, or $f\colon x \mapsto y$.
- Functions can be composed $X \xrightarrow{f} Y \xrightarrow{g} Z$, denoted $X \xrightarrow{f;g} Z$.



Functions

- A function from X to Y is an assignment:
  - Each $x \in X$ is assigned exactly one $y \in Y$.
  - We write $x \mapsto y$ and say x is *sent to* y.
- If the function is called f, we write $f\colon X \to Y$ or $X \xrightarrow{f} Y$,
  - and $f(x) = y$ means f sends x to y, or $f\colon x \mapsto y$.
- Functions can be composed $X \xrightarrow{f} Y \xrightarrow{g} Z$, denoted $X \xrightarrow{f;g} Z$.

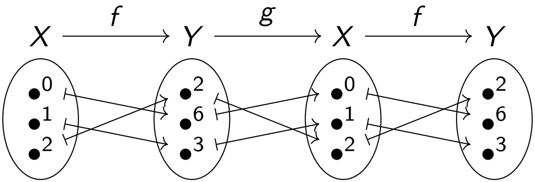


Identity and isomorphism

- For any set X, there is a function $\mathrm{id}_X \colon X \to X$.
    - It is defined by $\mathrm{id}_X(x) = x$ for any $x \in X$.
    - It's a unit for composition: $f ; \mathrm{id} = f$ and $\mathrm{id} ; f = f$.
- An *isomorphism* from X to Y is
    - a function $f \colon X \to Y$ and a function $g \colon Y \to X$
    - such that $f ; g = \mathrm{id}_X$ and $g ; f = \mathrm{id}_Y$.
- Isomorphisms are just a formal renaming of dots.

Identity and isomorphism

- For any set X, there is a function $\mathrm{id}_X\colon X \to X$.
  - It is defined by $\mathrm{id}_X(x) = x$ for any $x \in X$.
  - It's a unit for composition: $f;\mathrm{id} = f$ and $\mathrm{id};f = f$.
- An *isomorphism* from X to Y is
  - a function $f\colon X \to Y$ and a function $g\colon Y \to X$
  - such that $f;g = \mathrm{id}_X$ and $g;f = \mathrm{id}_Y$.
- Isomorphisms are just a formal renaming of dots.



Cantor's definition of cardinality: isomorphism

- So what is the cardinality $|X|$ of a finite set X?
  - Its the $n \in \mathbb{N}$ such that $X \cong \{1, 2, \ldots, n\}$.
  - We recover cardinality from the notion of isomorphism.
- For arbitrary A and B, how many isomorphisms $A \to B$ are there?
  - If A and B have different number of elements?

Cantor's definition of cardinality: isomorphism

- So what is the cardinality $|X|$ of a finite set X?
  - Its the $n \in \mathbb{N}$ such that $X \cong \{1, 2, \ldots, n\}$.
  - We recover cardinality from the notion of isomorphism.
- For arbitrary A and B, how many isomorphisms $A \to B$ are there?
  - If A and B have different number of elements? None!
  - If they both have n elements?

Cantor's definition of cardinality: isomorphism

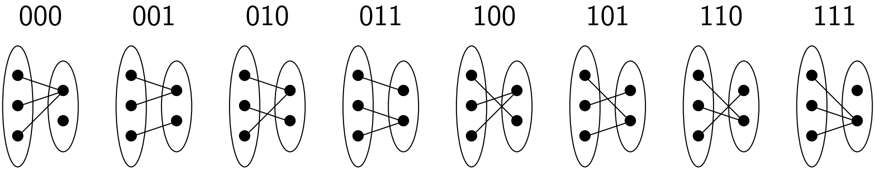
- So what is the cardinality $|X|$ of a finite set X?
  - Its the $n \in \mathbb{N}$ such that $X \cong \{1, 2, \ldots, n\}$.
  - We recover cardinality from the notion of isomorphism.
- For arbitrary A and B, how many isomorphisms $A \to B$ are there?
  - If A and B have different number of elements? None!
  - If they both have n elements? $n!$.
- Several more slides about relatively straightforward math.
  - Goal: show that sets and functions are primary.
  - Numerical operations are secondary, following from a deeper story.

How many functions from A to B?

- How many functions are there from $\underline{3}$ to $\underline{2}$?

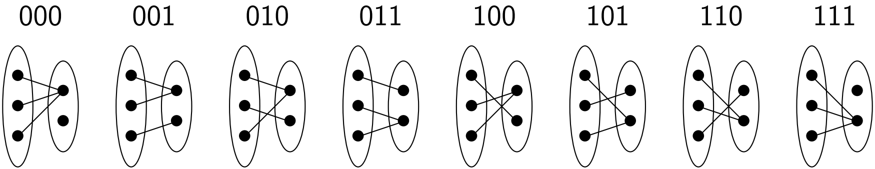
How many functions from A to B?

- How many functions are there from $\underline{3}$ to $\underline{2}$?



How many functions from A to B?

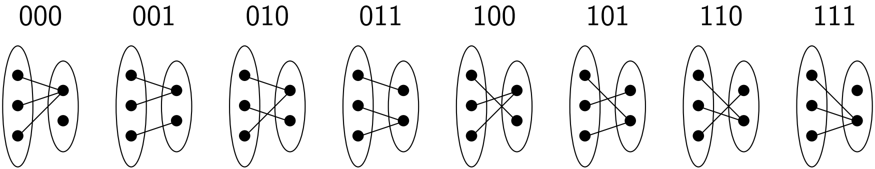
- How many functions are there from $\underline{3}$ to $\underline{2}$?



- In general there are n^m functions $\underline{m} \to \underline{n}$.
- Does it always work?
  - Are there really 1^m functions $\underline{m} \to \underline{1}$?
  - Are there really n^0 functions $\underline{0} \to \underline{n}$?
  - Are there really n^1 functions $\underline{1} \to \underline{n}$?

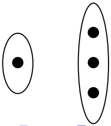
How many functions from A to B?

- How many functions are there from $\underline{3}$ to $\underline{2}$?



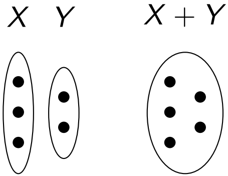
- In general there are n^m functions $\underline{m} \to \underline{n}$.
- Does it always work?
  - Are there really 1^m functions $\underline{m} \to \underline{1}$?
  - Are there really n^0 functions $\underline{0} \to \underline{n}$?
  - Are there really n^1 functions $\underline{1} \to \underline{n}$?

$\underline{1} \longrightarrow \underline{n}$



Disjoint unions

The disjoint union of sets X and Y is given by lumping them together.

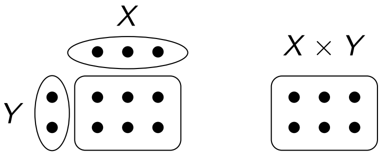


- Disjoint union, also called *coproduct*, is an operation on pairs of sets.
  - For any sets X and Y, we have a new set $X + Y$.
  - It categorifies the $+$ operation on natural numbers.
  - It is associative and commutative, and $\underline{0}$ is a unit.

$$X+Y \cong Y+X \qquad X+(Y+Z) \cong (X+Y)+Z \qquad X+\underline{0} \cong X \cong \underline{0}+X$$

Products

The *product* of sets X and Y is given by making a grid.



- Product is an operation on pairs of sets.
  - For any sets X and Y, we have a new set $X \times Y$.
  - It categorifies the $\times$ operation on natural numbers.
  - It is associative and commutative, and $\underline{1}$ is a unit.
  - It "distributes over addition" in the usual way.

$$X \times Y \cong Y \times X \qquad X \times (Y \times Z) \cong (X \times Y) \times Z \qquad X \times \underline{1} \cong X \cong \underline{1} \times X$$

$$X \times (Y + Z) \cong (X \times Y) + (X \times Z) \qquad X \times \underline{0} \cong \underline{0} \cong \underline{0} \times X$$

What more does arithmetic tell us?

We now have categorified 0, 1, $+$, $\times$, and ˆ.

- Category theory says that mappings govern all. So ˆ is paramount.
  - The fact that $1^n \cong 1$ means there is exactly one map $\underline{n} \to \underline{1}$.
  - The fact that $n^0 \cong 1$ means there is exactly one map $\emptyset \to \underline{n}$.
  - These *define* 0 and 1 in any category!
- So what do the following arithmetic facts tell us?

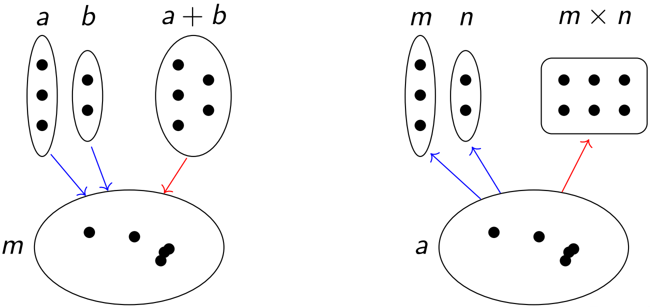
$$m^a \times m^b = m^{a+b} \qquad\qquad m^a \times n^a = (m \times n)^a$$

What more does arithmetic tell us?

We now have categorified 0, 1, $+$, $\times$, and ^.

- Category theory says that mappings govern all. So ^ is paramount.
  - The fact that $1^n \cong 1$ means there is exactly one map $\underline{n} \to \underline{1}$.
  - The fact that $n^0 \cong 1$ means there is exactly one map $\emptyset \to \underline{n}$.
  - These *define* 0 and 1 in any category!
- So what do the following arithmetic facts tell us?

$$m^a \times m^b = m^{a+b} \qquad m^a \times n^a = (m \times n)^a$$



So... is this going to be a pure math talk?

"That's all well and good, but the last time I worked on bags of dots...."

So... is this going to be a pure math talk?

"That's all well and good, but the last time I worked on bags of dots...."

- Pure math is like art in that it is beloved for its own sake.
- But it's unlike art in that it is valued for it's technical contribution.
- Working scientists and engineers want math they can use.

So... is this going to be a pure math talk?

"That's all well and good, but the last time I worked on bags of dots...."

- Pure math is like art in that it is beloved for its own sake.
- But it's unlike art in that it is valued for it's technical contribution.
- Working scientists and engineers want math they can use.

So what can you *use* about category theory (CT)?

- CT isolates the common themes that exist throughout pure math.
  - "One ring to rule them all" sounds pretty good. Explain it again.

So... is this going to be a pure math talk?

"That's all well and good, but the last time I worked on bags of dots...."

- Pure math is like art in that it is beloved for its own sake.
- But it's unlike art in that it is valued for it's technical contribution.
- Working scientists and engineers want math they can use.

So what can you *use* about category theory (CT)?

- CT isolates the common themes that exist throughout pure math.
  - "One ring to rule them all" sounds pretty good. Explain it again.
  - CT says: Understanding something is knowing how it relates.
- Is this just a slogan? Is this buddhist postmodernist hippy nonsense?
  - No, it's called "abstract nonsense": concrete essences abstracted away.

So... is this going to be a pure math talk?

"That's all well and good, but the last time I worked on bags of dots...."

- Pure math is like art in that it is beloved for its own sake.
- But it's unlike art in that it is valued for it's technical contribution.
- Working scientists and engineers want math they can use.

So what can you *use* about category theory (CT)?

- CT isolates the common themes that exist throughout pure math.
  - "One ring to rule them all" sounds pretty good. Explain it again.
  - CT says: Understanding something is knowing how it relates.
- Is this just a slogan? Is this buddhist postmodernist hippy nonsense?
  - No, it's called "abstract nonsense": concrete essences abstracted away.
  - But it's not nonsense: it's caused a paradigm shift within math.
  - And you can see for yourself: let's recap and then move on.

Recapping

The take-away so far is that:

- Numbers $0, 1, 2, \ldots$ are shadows on Plato's cave.
  - Behind them lurk finite sets $\{\}, \{1\}, \{a, b\}$.
  - The finite *set* of people in this room is far richer than the *number*.
- The constants $0, 1$ and operations $+, \times, \hat{}$ are also shadows.
  - They are completely defined by relationships, e.g. $m^{a+b} = m^a \times m^b$.
  - Every set is completely determined by the maps into and out of it.

So what is categorification?

Recapping

The take-away so far is that:

- Numbers $0, 1, 2, \ldots$ are shadows on Plato's cave.
  - Behind them lurk finite sets $\{\}, \{1\}, \{a, b\}$.
  - The finite *set* of people in this room is far richer than the *number*.
- The constants $0, 1$ and operations $+, \times, \hat{}$ are also shadows.
  - They are completely defined by relationships, e.g. $m^{a+b} = m^a \times m^b$.
  - Every set is completely determined by the maps into and out of it.

So what is categorification?

It's searching for the relationships that explain the shadows in Plato's cave.

Some other categories

A category is roughly a bunch of objects and relationships.

- We just discussed the category with finite sets and functions.
- There is also the category of vector spaces and linear transformations.
  - What are the "cardinalities" in this case? $\mathbb{R}^n$ takes the place of $\underline{n}$.
  - The isomorphisms are the invertible matrices.
  - One can take products and sums of vector spaces.
- There is the category of graphs and graph homomorphisms.
- There is the category of databases and database mappings.
- There is the category of data types and programs.

In each case the above story can be found hidden inside.

Some other categories

A category is roughly a bunch of objects and relationships.

- We just discussed the category with finite sets and functions.
- There is also the category of vector spaces and linear transformations.
  - What are the "cardinalities" in this case? $\mathbb{R}^n$ takes the place of $\underline{n}$.
  - The isomorphisms are the invertible matrices.
  - One can take products and sums of vector spaces.
- There is the category of graphs and graph homomorphisms.
- There is the category of databases and database mappings.
- There is the category of data types and programs.

In each case the above story can be found hidden inside.

- And then there are custom categories.

Outline

Let's learn about shale (schiste argileux)

As scientists, sometimes we have to learn about an unfamiliar subject.

- Familiarity with a subject is...

Let's learn about shale (schiste argileux)

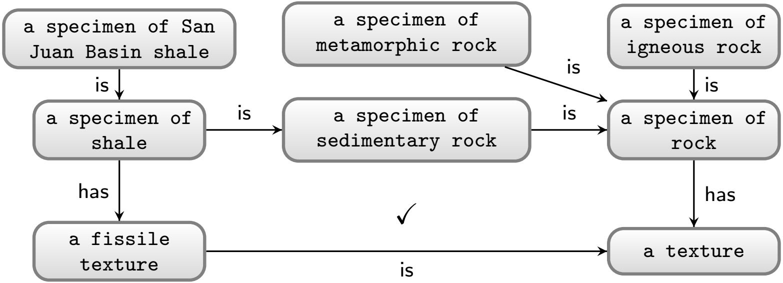
As scientists, sometimes we have to learn about an unfamiliar subject.

- Familiarity with a subject is... knowing the ideas and how they relate.
- I don't know anything about shale, not much more about Télécom.
- Let's talk about shale a little bit, just to consider the unfamiliar.

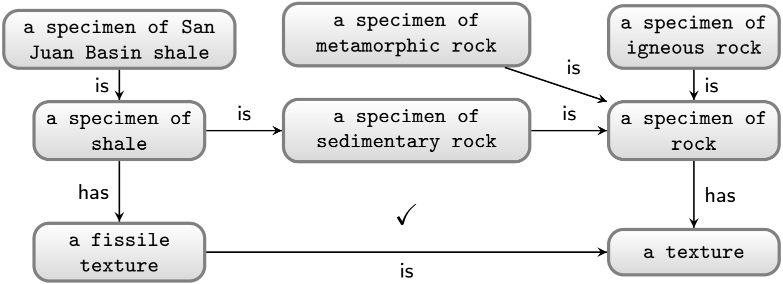
Let's learn about shale (schiste argileux)

As scientists, sometimes we have to learn about an unfamiliar subject.

- Familiarity with a subject is... knowing the ideas and how they relate.
- I don't know anything about shale, not much more about Télécom.
- Let's talk about shale a little bit, just to consider the unfamiliar.

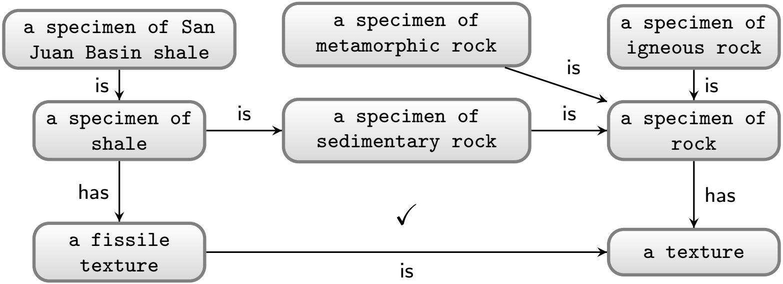


Is this just a concept web?



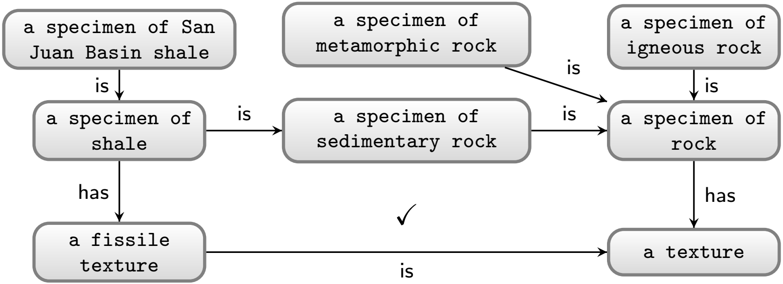
Concept webs can be useful. But are they math?

Is this just a concept web?



- One way to make this mathematical is to require that:
  - every box can be assigned a finite set of examples.
  - every arrow can be assigned a function between the finite sets.
  - the two functions ⌜shale⌝ $\to$ ⌜texture⌝ should agree (hence the ✓).

Is this just a concept web?



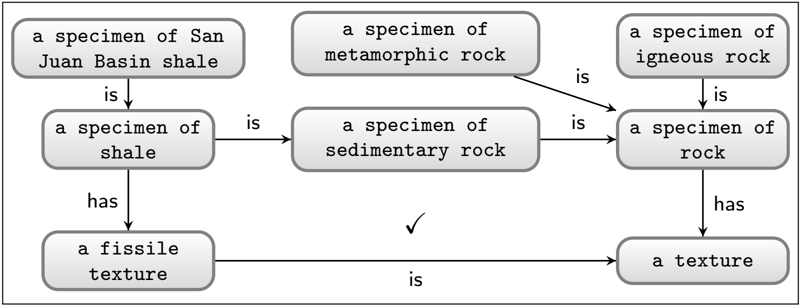
- One way to make this mathematical is to require that:
  - every box can be assigned a finite set of examples.
  - every arrow can be assigned a function between the finite sets.
  - the two functions $\ulcorner\text{shale}\urcorner \to \ulcorner\text{texture}\urcorner$ should agree (hence the ✓).
- Perhaps also that there is a coproduct:

$$\ulcorner\text{igneous}\urcorner + \ulcorner\text{metamorphic}\urcorner + \ulcorner\text{sedimentary}\urcorner \cong \ulcorner\text{rock}\urcorner.$$

A functor connects our schema to data

We have built a custom category:

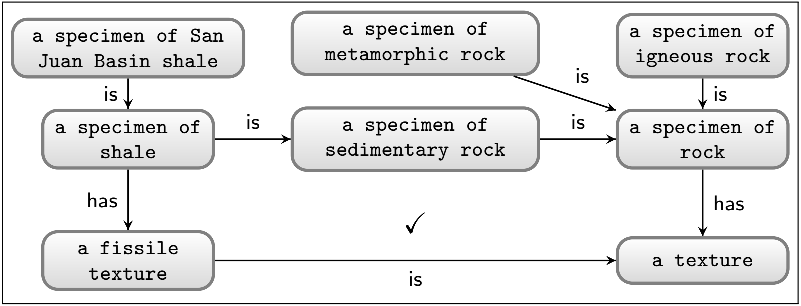
$$\mathcal{C} =$$



A functor connects our schema to data

We have built a custom category:

$\mathcal{C} =$



and we give it meaning by associating a functor $\mathcal{C} \to$ **FinSet**.

Let's back up a little

- I want to define categories and functors so that this makes sense.
- Then we'll have our whirlwind tour of examples.

Outline

A category is a graph with extra structure

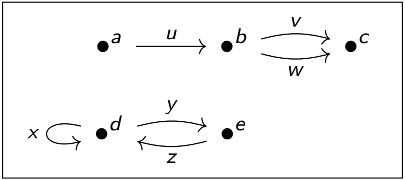
Recall that a graph (a.k.a. directed multigraph) consists of

- A bunch of dots $\bullet$, and a bunch of arrows $\bullet^a \to \bullet^b$.
- There can be several arrows $\bullet^a \rightrightarrows \bullet^b$, and even arrows $\bullet^a \to \bullet^a$.
- A path is a sequence $\bullet^{a_0} \to \bullet^{a_1} \to \cdots \to \bullet^{a_n}$.
  - A path can have any length $n \in \mathbb{N}$ including $n = 0$ and $n = 1$.

A category is a graph with extra structure

Recall that a graph (a.k.a. directed multigraph) consists of

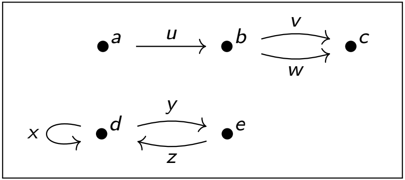
- A bunch of dots $\bullet$, and a bunch of arrows $\bullet^a \to \bullet^b$.
- There can be several arrows $\bullet^a \rightrightarrows \bullet^b$, and even arrows $\bullet^a \to \bullet^a$.
- A path is a sequence $\bullet^{a_0} \to \bullet^{a_1} \to \cdots \to \bullet^{a_n}$.
  - A path can have any length $n \in \mathbb{N}$ including $n = 0$ and $n = 1$.



A category is a graph with extra structure

Recall that a graph (a.k.a. directed multigraph) consists of

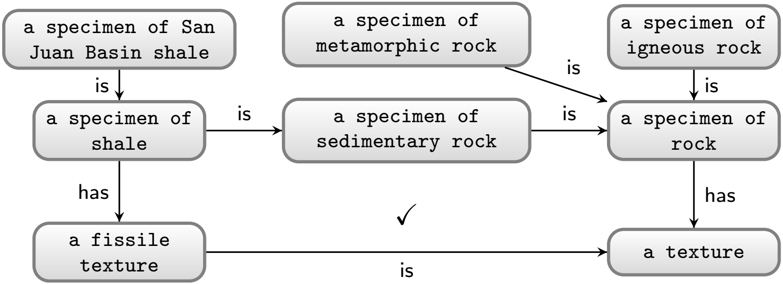
- A bunch of dots $\bullet$, and a bunch of arrows $\bullet^a \to \bullet^b$.
- There can be several arrows $\bullet^a \rightrightarrows \bullet^b$, and even arrows $\bullet^a \to \bullet^a$.
- A path is a sequence $\bullet^{a_0} \to \bullet^{a_1} \to \cdots \to \bullet^{a_n}$.
  - A path can have any length $n \in \mathbb{N}$ including $n = 0$ and $n = 1$.



A category is a graph *plus* the ability to declare two paths equivalent.

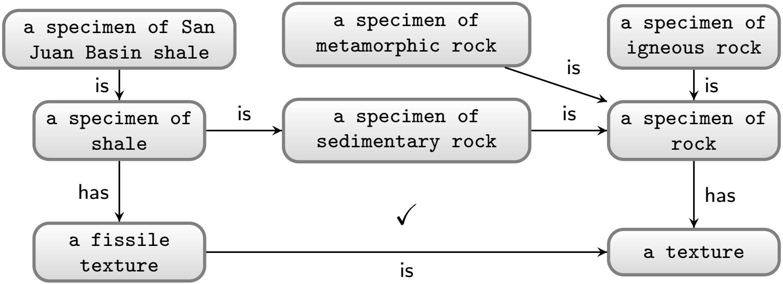
- So we could declare $u; v \simeq u; w$ and/or $x; x; x \simeq \mathrm{id}_d$.
- The concatenations of equivalent paths are equivalent.

Back to the shale category



- This graph happens to have eight vertices and eight arrows.
  - Here we know that every rock has a texture,
  - some textures are fissile, shale is a rock, etc.

Back to the shale category



- This graph happens to have eight vertices and eight arrows.
  - Here we know that every rock has a texture,
  - some textures are fissile, shale is a rock, etc.
- To make it a category we can add any path equivalences we want.
  - We want to say that shale's texture is fissile.
  - It follows that the texture of San Juan shale is also fissile.

The category of finite sets

We said that there is a category **FinSet** of finite sets.

- There is a vertex (usually called an *object*) for each finite set A.

$$\bullet^{\underline{17}} \qquad \bullet^{\{a,b,c\}}$$

- There is an arrow (also called a *morphism*) for each function $f\colon A \to B$.
- Two paths are equivalent if their composite is the same function.

The category of finite sets

We said that there is a category **FinSet** of finite sets.

- There is a vertex (usually called an *object*) for each finite set A.

$$\bullet^{\underline{17}} \qquad \bullet^{\{a,b,c\}}$$

- There is an arrow (also called a *morphism*) for each function $f\colon A \to B$.
- Two paths are equivalent if their composite is the same function.

Just to be clear, **FinSet** is a huge category.

- There are 871^{23} arrows from vertex $\underline{23}$ to vertex $\underline{871}$.
- There are many paths from $\underline{23}$ to $\underline{1}$, but they are all equivalent.

A functor is a sort of graph mapping

- Suppose we have two categories $\mathcal{C}$ and $\mathcal{D}$.
- We want to connect them by a functor $F\colon \mathcal{C} \to \mathcal{D}$.
- A functor is like a function, except it has to deal with more structure.
  - F sends the vertices in $\mathcal{C}$ to the vertices in $\mathcal{D}$.
  - F sends the arrows in $\mathcal{C}$ to *paths* in $\mathcal{D}$.
  - So a path p in $\mathcal{C}$ is also sent to a path $F(p)$ in $\mathcal{D}$.
  - And if $p \simeq q$ are equivalent paths in $\mathcal{C}$, then $F(p) \simeq F(q)$ in $\mathcal{D}$.

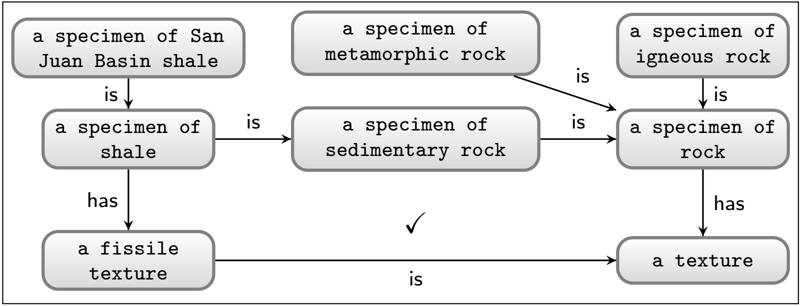
A functor is a sort of graph mapping

- Suppose we have two categories $\mathcal{C}$ and $\mathcal{D}$.
- We want to connect them by a functor $F\colon \mathcal{C} \to \mathcal{D}$.
- A functor is like a function, except it has to deal with more structure.
  - F sends the vertices in $\mathcal{C}$ to the vertices in $\mathcal{D}$.
  - F sends the arrows in $\mathcal{C}$ to *paths* in $\mathcal{D}$.
  - So a path p in $\mathcal{C}$ is also sent to a path $F(p)$ in $\mathcal{D}$.
  - And if $p \simeq q$ are equivalent paths in $\mathcal{C}$, then $F(p) \simeq F(q)$ in $\mathcal{D}$.

- So a functor $\mathcal{C} \to$ **FinSet** consists of:
  - A finite set $F(c)$ for every vertex $c \in \mathcal{C}$,
  - a function $F(c) \to F(d)$ for every arrow $c \to d$ in $\mathcal{D}$,
  - and if two paths are equivalent $p \simeq q$ in $\mathcal{C}$, then $F(p)$ and $F(q)$ must be the same function.

Shale data

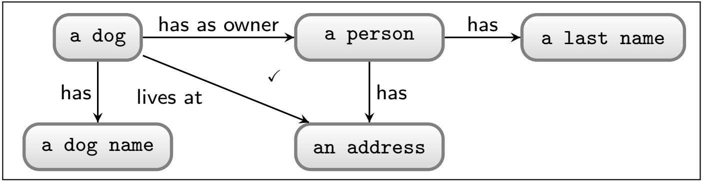
$\mathcal{S} =$



What is a functor $\mathcal{S} \to$ **FinSet**?

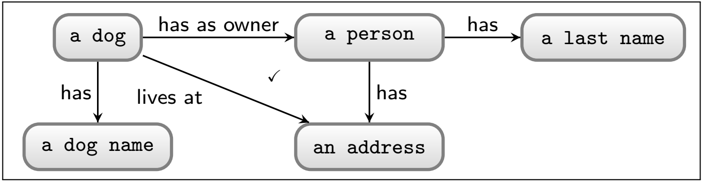
- Imagine the set of all shale specimens and that of all rock specimens.
- Imagine is a function $\ulcorner$shale$\urcorner \to \ulcorner$rock$\urcorner$.
- Imagine the set of all textures and that of all fissile textures.
- Imagine the rest of the structure....

Custom categories are database schemas

$\mathcal{D} =$ 

Custom categories are database schemas

$\mathcal{D} =$

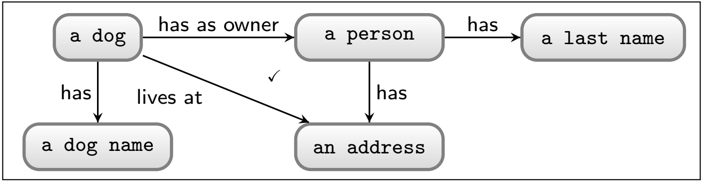


$\xrightarrow{F}$ **FinSet**

dog			
ID	**name**	**owner**	**address**
D101	Wally	P34	15 Ash St.
D102	Fido	P46	201 5th Ave.
D104	Buster	P17	27 Spring Ln.

person		
ID	**lastName**	**address**
P17	Jones	27 Spring Ln.
P19	Smith	201 Gladys Ave.
P34	Smith	15 Ash St.
P46	D'Angelo	201 5th Ave.

Custom categories are database schemas

$\mathcal{D} =$  $\xrightarrow{F}$ **FinSet**

dog			
ID	**name**	**owner**	**address**
D101	Wally	P34	15 Ash St.
D102	Fido	P46	201 5th Ave.
D104	Buster	P17	27 Spring Ln.

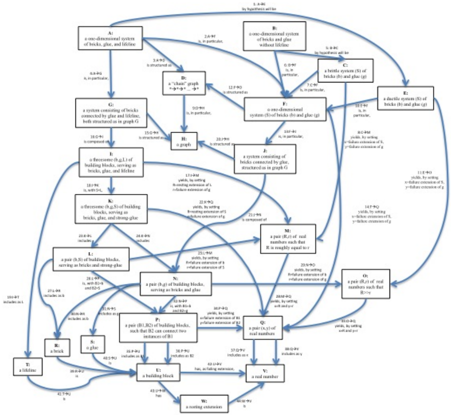
person		
ID	**lastName**	**address**
P17	Jones	27 Spring Ln.
P19	Smith	201 Gladys Ave.
P34	Smith	15 Ash St.
P46	D'Angelo	201 5th Ave.

Note that equivalent paths are assigned the same composite function.

Outline

An olog from materials science

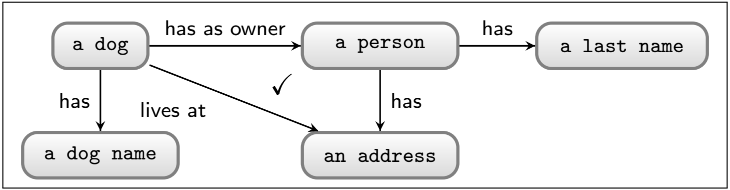
I call custom categories "ologs". Here's one from materials science.[1]



[1] Spivak, D.I.; Giesa, T.; Wood, E.; Buehler, M.J. (2011) "Category Theoretic Analysis of Hierarchical Protein Materials and Social Networks." *PLoS ONE* 6(9): e23911. doi:10.1371/journal.pone.0023911

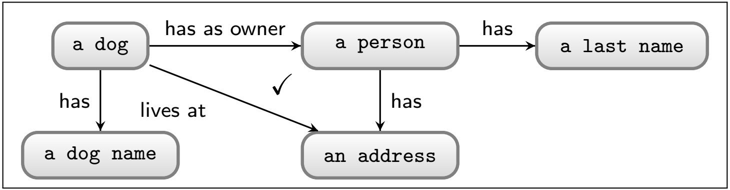
Communicating between conceptualizations

Think of an olog as a conceptualization of something.



Communicating between conceptualizations

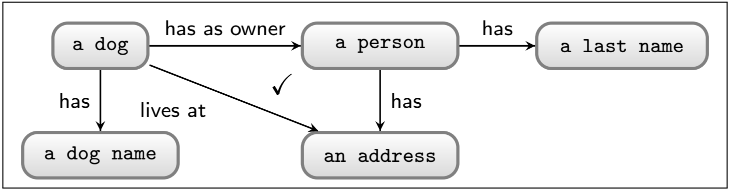
Think of an olog as a conceptualization of something.



- Formally it's a category; the labels are invisible to the math.
- We can compare two ologs with a functor:
  - If you have a olog/category $\mathcal{C}$ for your conceptualization,
  - and I have a olog/category $\mathcal{D}$ for my conceptualization,
  - there may be a meaningful functor $\mathcal{C} \to \mathcal{D}$ or $\mathcal{D} \to \mathcal{C}$
  - or an overlap $\mathcal{O}$ and meaningful functors $\mathcal{C} \leftarrow \mathcal{O} \to \mathcal{D}$.

Communicating between conceptualizations

Think of an olog as a conceptualization of something.



- Formally it's a category; the labels are invisible to the math.
- We can compare two ologs with a functor:
  - If you have a olog/category $\mathcal{C}$ for your conceptualization,
  - and I have a olog/category $\mathcal{D}$ for my conceptualization,
  - there may be a meaningful functor $\mathcal{C} \to \mathcal{D}$ or $\mathcal{D} \to \mathcal{C}$
  - or an overlap $\mathcal{O}$ and meaningful functors $\mathcal{C} \leftarrow \mathcal{O} \to \mathcal{D}$.
- But recall that filling an olog with data is also a functor $\mathcal{C} \to$ **FinSet**.

Moving data along functors

- Suppose $\mathcal{D}$ is a category and $F\colon \mathcal{D} \to \mathbf{FinSet}$ is a functor.
  - So $\mathcal{D}$ could be the dog olog and F is the data about Wally, etc.
- Suppose that $\mathcal{P}$ is another category and $i\colon P \to \mathcal{D}$ is a functor.
  - So $\mathcal{P}$ could be the person sub-olog, and i the inclusion.
- We can compose i and F to get a new functor

$$\mathcal{P} \xrightarrow{\ i\ } \mathcal{D} \xrightarrow{\ F\ } \mathbf{FinSet}, \qquad \mathcal{P} \xrightarrow{\ i;F\ } \mathbf{FinSet}$$

- So now olog $\mathcal{P}$ has data in it, borrowed from friend-olog $\mathcal{D}$ using i.

Functorial data migration

- Suppose given two conceptualizations, yours $\mathcal{C}$ and mine $\mathcal{D}$.
- Suppose we can translate between them with a functor $T\colon \mathcal{C} \to \mathcal{D}$.

Functorial data migration

- Suppose given two conceptualizations, yours $\mathcal{C}$ and mine $\mathcal{D}$.
- Suppose we can translate between them with a functor $T\colon \mathcal{C} \to \mathcal{D}$.
- Then in fact there are three data-migration functors, denoted:

$$\Pi_T\colon \mathcal{C}\text{-data} \to \mathcal{D}\text{-data}, \quad \Delta_T\colon \mathcal{D}\text{-data} \to \mathcal{C}\text{-data}, \quad \Sigma_T\colon \mathcal{C}\text{-data} \to \mathcal{D}\text{-data}.$$

Functorial data migration

- Suppose given two conceptualizations, yours $\mathcal{C}$ and mine $\mathcal{D}$.
- Suppose we can translate between them with a functor $T\colon \mathcal{C} \to \mathcal{D}$.
- Then in fact there are three data-migration functors, denoted:

$$\Pi_T\colon \mathcal{C}\text{-data} \to \mathcal{D}\text{-data}, \quad \Delta_T\colon \mathcal{D}\text{-data} \to \mathcal{C}\text{-data}, \quad \Sigma_T\colon \mathcal{C}\text{-data} \to \mathcal{D}\text{-data}.$$

- They move data from one conceptual schema to another.
- And they generalize *join, project, union* from database theory.
- So if there is even an overlap $\mathcal{C} \leftarrow \mathcal{O} \to \mathcal{D}$, we can move and share data.

Categories and functors everywhere

What to consider during the whirlwind tour:

- I'll discuss lots of different categories and functors.
- Each category is a conceptual reference frame.
  - An olog, say about dogs or shale, is a conceptual reference frame.
  - "What is the domain of discourse, and what have we agreed to?"
  - But even **FinSet** is a conceptual frame: sets, functions, $+$, $\times$, etc.

Categories and functors everywhere

What to consider during the whirlwind tour:

- I'll discuss lots of different categories and functors.
- Each category is a conceptual reference frame.
  - An olog, say about dogs or shale, is a conceptual reference frame.
  - "What is the domain of discourse, and what have we agreed to?"
  - But even **FinSet** is a conceptual frame: sets, functions, $+$, $\times$, etc.
- Each functor $\mathcal{C} \to \mathcal{D}$ is a connection between conceptual frames.
  - We can transport data along functors (because data is a functor too!)
  - We can transport problems from $\mathcal{C}$ to $\mathcal{D}$ and solve them there.
  - For example, in algebraic topology they analyze spaces using functors **Spaces** $\to$ **Groups**.

Categories and functors everywhere

What to consider during the whirlwind tour:

- I'll discuss lots of different categories and functors.
- Each category is a conceptual reference frame.
  - An olog, say about dogs or shale, is a conceptual reference frame.
  - "What is the domain of discourse, and what have we agreed to?"
  - But even **FinSet** is a conceptual frame: sets, functions, $+$, $\times$, etc.
- Each functor $\mathcal{C} \to \mathcal{D}$ is a connection between conceptual frames.
  - We can transport data along functors (because data is a functor too!)
  - We can transport problems from $\mathcal{C}$ to $\mathcal{D}$ and solve them there.
  - For example, in algebraic topology they analyze spaces using functors **Spaces** $\to$ **Groups**.

Let's start the tour.

Posets

Poset means partially ordered set $(P, \leq)$. Posets are simple categories.

- In a category, there may be many non-equivalent paths $A \to B$.
  - For example, there are 871^{23} non-equivalent paths $\underline{23} \to \underline{871}$.
  - Think of three non-equivalent arrows $\ulcorner\text{person}\urcorner \to \ulcorner\text{person}\urcorner$.
- A poset is a category with the guarantee that every two paths (having the same endpoints) are equivalent.

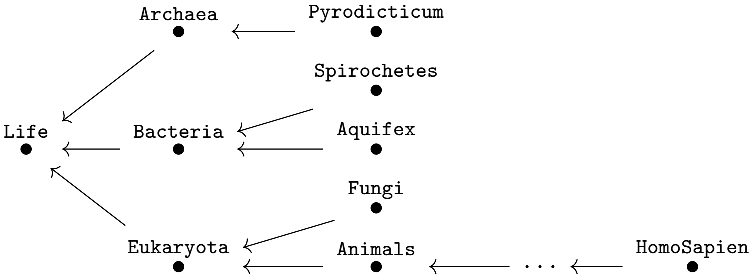
Posets

Poset means partially ordered set $(P, \leq)$. Posets are simple categories.

- In a category, there may be many non-equivalent paths $A \to B$.
  - For example, there are 871^{23} non-equivalent paths $\underline{23} \to \underline{871}$.
  - Think of three non-equivalent arrows $\ulcorner$person$\urcorner \to \ulcorner$person$\urcorner$.
- A poset is a category with the guarantee that every two paths (having the same endpoints) are equivalent.
- For example, consider the poset $(\mathbb{R}, \leq)$ of real numbers.
  - As a category, its objects are all real numbers, $0, 1, \pi, -4.11$, etc.
  - There is (exactly one) morphism $a \to b$ iff $a \leq b$.
  - A path $a \to \cdots \to b$ just means $a \leq \cdots \leq b$, meaning $a \leq b$.

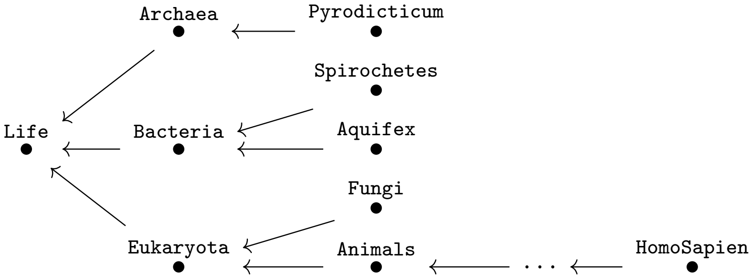
Hierarchies

- A hierarchy is just a poset, and as such is a simple category.



Hierarchies

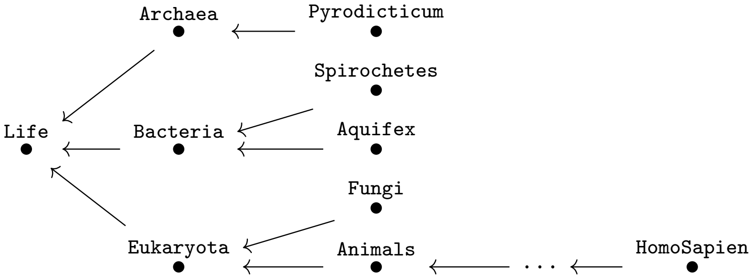
- A hierarchy is just a poset, and as such is a simple category.



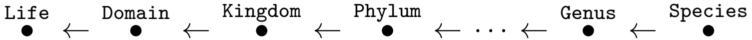
- Functors between posets are the usual "monotonic functions".
- We also have the poset below and a functor from top to bottom.

Hierarchies

- A hierarchy is just a poset, and as such is a simple category.



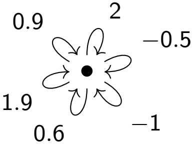
- Functors between posets are the usual "monotonic functions".
- We also have the poset below and a functor from top to bottom.



Monoids

Monoids are also simple categories, but in the opposite way.

- A *monoid* is a category with one object ("mono"=1).
- Consider the set $\mathbb{R}$ of real numbers and the multiplication operation $*$.
- These together can be considered as a category $\mathcal{R}$:
  - $\mathcal{R}$ has one object, call it $\bullet$.
  - It has an arrow $\bullet \xrightarrow{x} \bullet$ for each real number $x \in \mathbb{R}$.
  - A path is just $\bullet \xrightarrow{x_1} \cdots \xrightarrow{x_n} \bullet$.
  - We make it equivalent to its product, the one-arrow path $x_1 * \cdots * x_n$.



Actions

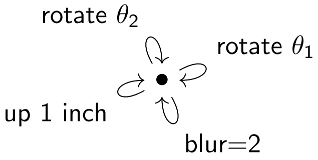
Consider a paint program, and let P be the set of drawable pictures.

- What actions can you take?
  - You can take any picture $p \in P$ and rotate it by an angle θ.
  - You can move any picture $p \in P$ up or down on the canvass.
  - You can fill up a closed curve in p with color.
  - Once done, you get a new picture $p' \in P$.
- These actions are the arrows in a monoid.

Actions

Consider a paint program, and let P be the set of drawable pictures.

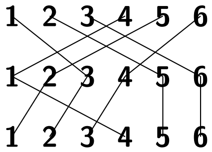
- What actions can you take?
  - You can take any picture $p \in P$ and rotate it by an angle θ.
  - You can move any picture $p \in P$ up or down on the canvass.
  - You can fill up a closed curve in p with color. (always?)
  - Once done, you get a new picture $p' \in P$.
- These actions are the arrows in a monoid.
  - The "one-object" assumption says you can do any sequence of actions.
  - A path is a sequence of actions you can perform on "my pic".
  - Two paths are equal if they always act the same
  - For example rotating θ_1 then θ_2 is rotating $\theta_1 + \theta_2$.



Groups

Groups are monoids (one-object categories) where every morphism is iso.

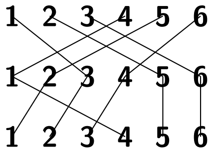
- Recall that there are 720 isomorphisms from $\underline{6}$ to itself.
- These form a monoid: a path of isomorphisms is an isomorphism.



Groups

Groups are monoids (one-object categories) where every morphism is iso.

- Recall that there are 720 isomorphisms from $\underline{6}$ to itself.
- These form a monoid: a path of isomorphisms is an isomorphism.



- A group is a monoid with the guarantee that every map is invertible.
- Moving a glass around is invertible, but smashing it isn't.
- Every action on a picture is invertible, except emailing it to a friend.

A group is a system of symmetries.

Sheaves

Sheaves were the original killer app for categories.

- Of course, we're talking "killer pure math app".
  - A. Grothendieck used sheaves to prove conjectures in number theory.
  - It completely transformed algebraic geometry, and other fields followed.
- Sheaves are for piecing together locally-gathered data. Examples:
  - Temperature data is gathered locally, producing worldwide coverage.
  - Security camera footage, though local, can be pieced together.
- We talk about sheaves "on a space" X, e.g. on earth's surface.

The category of all...

- One can also consider the category of all
  - finite sets,
  - groups,
  - monoids,
  - graphs,
  - posets,
  - spaces,
  - sheaves on X.
  - ologs/categories,
  - etc.

The category of all...

- One can also consider the category of all
  - finite sets,
  - groups,
  - monoids,
  - graphs,
  - posets,
  - spaces,
  - sheaves on X.
  - ologs/categories,
  - etc.
- There are lots and lots of functors between them, e.g. those that
  - forget structure: **Grp** $\to$ **Mon** and **Mon** $\to$ **FinSet**;
  - freely add structure: **FinSet** $\to$ **Mon** and **Mon** $\to$ **Grp**; or
  - much more complex things like Fields medal functors **Shv**$(X) \to$ **Grp**.

The category of all...

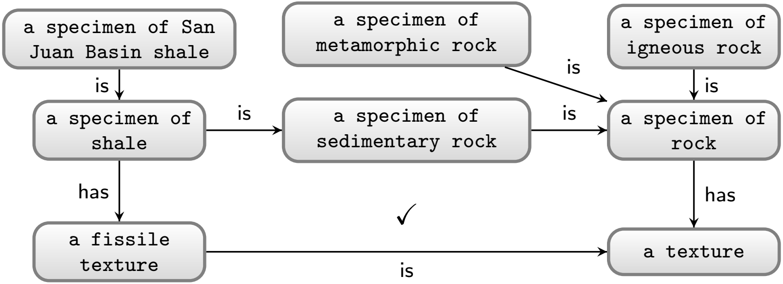
- One can also consider the category of all
  - finite sets,
  - groups,
  - monoids,
  - graphs,
  - posets,
  - spaces,
  - sheaves on X.
  - ologs/categories,
  - etc.
- There are lots and lots of functors between them, e.g. those that
  - forget structure: $\mathbf{Grp} \to \mathbf{Mon}$ and $\mathbf{Mon} \to \mathbf{FinSet}$;
  - freely add structure: $\mathbf{FinSet} \to \mathbf{Mon}$ and $\mathbf{Mon} \to \mathbf{Grp}$; or
  - much more complex things like Fields medal functors $\mathbf{Shv}(X) \to \mathbf{Grp}$.
- All of math is interconnected by functors, by which we can transfer problems and solutions.

Scratching the surface

Another analogy: The justice this talk does to category theory is kinda like...

Scratching the surface

Another analogy: The justice this talk does to category theory is kinda like...



...the justice the above olog does to shale.

Summary

What have we discussed?

- There's been a paradigm shift in math.
  - Old ideas have been categorified to show an underlying richness.
  - Subdisciplines have been connected into a rich web, using functors.
  - Different categories are fine-tuned for solving different problems.
  - Functors can transport problems and solutions between domains.

Summary

What have we discussed?

- There's been a paradigm shift in math.
  - Old ideas have been categorified to show an underlying richness.
  - Subdisciplines have been connected into a rich web, using functors.
  - Different categories are fine-tuned for solving different problems.
  - Functors can transport problems and solutions between domains.
- But the same abstractions also have applications to science.
  - A programming language is a category.
  - Custom categories can be connected by functorial bridges.
  - Actions, symmetries, hierarchies, etc. all captured by this framework.
  - Data is a functor, and it can be transported by functors.

Summary

What have we discussed?

- There's been a paradigm shift in math.
  - Old ideas have been categorified to show an underlying richness.
  - Subdisciplines have been connected into a rich web, using functors.
  - Different categories are fine-tuned for solving different problems.
  - Functors can transport problems and solutions between domains.
- But the same abstractions also have applications to science.
  - A programming language is a category.
  - Custom categories can be connected by functorial bridges.
  - Actions, symmetries, hierarchies, etc. all captured by this framework.
  - Data is a functor, and it can be transported by functors.

The future will be increasingly interdisciplinary. Category theory can help.

Summary

What have we discussed?

- There's been a paradigm shift in math.
  - Old ideas have been categorified to show an underlying richness.
  - Subdisciplines have been connected into a rich web, using functors.
  - Different categories are fine-tuned for solving different problems.
  - Functors can transport problems and solutions between domains.
- But the same abstractions also have applications to science.
  - A programming language is a category.
  - Custom categories can be connected by functorial bridges.
  - Actions, symmetries, hierarchies, etc. all captured by this framework.
  - Data is a functor, and it can be transported by functors.

The future will be increasingly interdisciplinary. Category theory can help.

Thanks for the opportunity to speak!