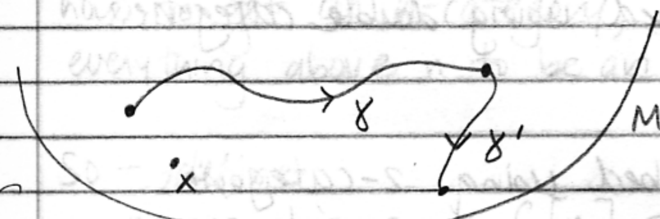
A - functor from paths to group

Baez seminar 1/22/02



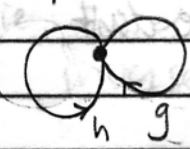
Gauge Theory

Want $A(\gamma) \in G$ G some group "gauge grp"

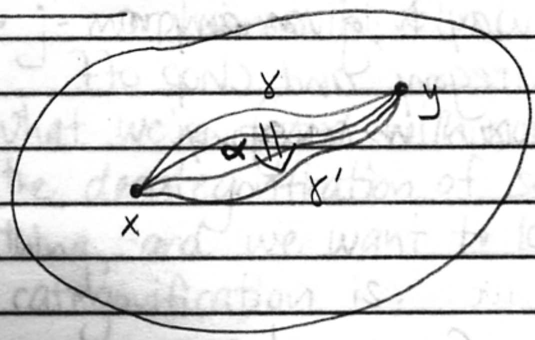
$A(\gamma)A(\gamma') = A(\gamma \cdot \gamma')$
$A(1_x) = 1 \in G$

category: objects are points, paths are morphisms

We can think of a group as a category:
objects — 1 object
morphisms — maps from elt to itself
(all morphisms are invertible)



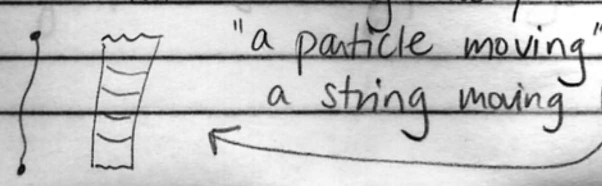
2 category (paths between paths)

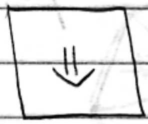


$\alpha : \gamma \Rightarrow \gamma'$

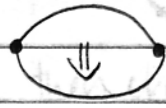
G will be a "2-group" - a 2-category w/ one object and all morphisms and 2 morphisms are equivalences.

We need this if string theory is actually true — "a particle moving" is really a string moving like



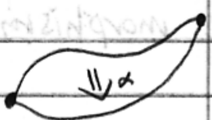


described using double categories



described using 2-categories

Recall — Any topological space X gives an ω-category $\Pi(X)$ where
- objects are pts. of X
- morphisms are paths in X



- (α) 2-morphisms are paths of paths

$\vdots$ etc

If we're studying 2-categories, we don't need the etc.

<u>Note</u>: We have a way to, given any n, take an ω-category and chop off top part to get an n-category.

<u>Decategorification</u>:

From an ω-category C we get an n-category $C[n]$ with:

objects in $C[n]$ = objects in C
morphisms in $C[n]$ = morphisms in C

note: working up to equivalences gives us eqns at upper levels

now to get an n-category, we need everything above n to be an identity.

So - define

- n-morphisms in $C[n]$ = equivalence classes of n-morphisms in C.

where given 2 n-morphisms

here we take equiv. classes

$f, g : x \longrightarrow y$ in C, we say $f \sim g$ f is <u>equivalent</u> to g if there is an equivalence $\alpha : f \Rightarrow g$

(ie, $\exists$ $\alpha^{-1} : g \Rightarrow f$ that's not a true inverse, but inverse up to equivalence)

- j-morphisms in $C[n]$ $j > n$ are identities

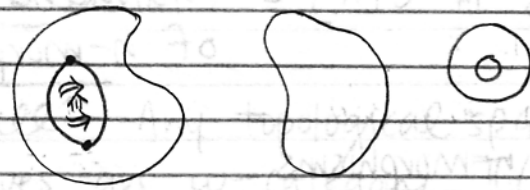
What we're <u>faced with</u> now - we're handed the decategorification of some mathematical thing, and we want to know what the categorification is.

Recall in $\pi(X)$, all morphisms are equivalences

<u>Fact</u>: <u>If</u> C is an ω-groupoid (all j-morphisms are equivalences) then $C[n]$ is an n-groupoid (n-cat w/ all j-morphisms equivalences).

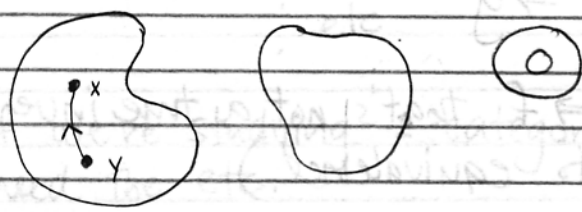
So: $\Pi(X)$ gives by decategorification
an n-groupoid: $\Pi_n(X)$.

$n=0$ $\quad \Pi_0(X)$ is a set



A zero morphism is an object

we form equiv classes of pts where 2 points are equivalent if connected by a path.



equivalence classes of points, where $x \sim y$ if there's a path from x to y.
We call this the set of <u>path-components</u>.

Decategorification — we only care if we <u>can</u> get from x to y, we don't care how (or how many ways) note: working up to equivalences gives us eqns at upper levels

(fundamental grpoid)

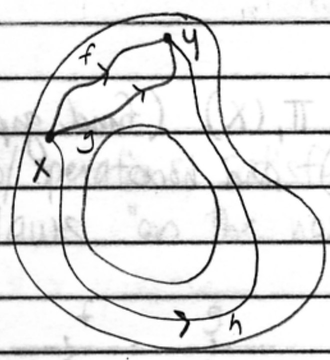
$n=1$ $\Pi_1(X)$ is a groupoid (no base pt for groupoid)

ie) A category where all morphisms have inverses.

- objects - points of X
- morphisms - homotopy classes of paths (we regard $f \sim g$ if can get to one from other thru deformation/homotopy)

here we take equiv. classes

composition must be associative on the nose



$f \not\sim h$

Fundamental group (only consider loops at x)

$$\pi_1(X, x) = \mathrm{hom}(x, x)$$

the fund grp is a microcosm of the fund. groupoid (only restrict ourselves to loops at a pt)

$\Pi_1(X)$ is called the fundamental groupoid.

$n=2$ $\quad \Pi_2(X)$ is a 2-groupoid.

Ex)  S^2 $\quad \alpha : 1_x \Rightarrow 1_x$

this path of paths tells us there is a 2-dim'l hole.

So $\alpha \neq 1_{1_x}$

Goal: Tell what $\Pi_n(X)$ tells us about our space.

$\Pi_2(X)$ includes the info of $\pi_1(X)$ (fund. grp) and $\pi_2(X)$ (homotopy class) and how they relate.

To-Do List

1) Π_2 is not right for categorified gauge theory
we need "thin Π_2"

2) Understand 2-groupoids:
strict vs. weak

laws hold on the nose

up to $\underline{\text{something}}$

3) Understand 2-groups: A 2-groupoid w/ only 1 object.

Just as a group was a groupoid (cat. w/ all morphisms equivalences) w/ one object,
define a 2-group to be a 2-groupoid w/ one object.
Again we have strict vs. weak.

Strict 2-categories:

A strict 2-category C is:
- a set C_0 of objects
- for any $x, y \in C_0$ a set of morphisms $f: x \to y$
- for any morphisms $f, g: x \to y$ a set of 2 morphisms $\alpha: f \Rightarrow g$

w/ operations satisfying various laws "on the nose":

① $\xrightarrow{f} \bullet \xrightarrow{g}$ gives composition of morphisms $f: x \to y$, $g: y \to z$ gives $fg: x \to z$

② $\bullet$ gives identities: x gives $1_x: x \to x$

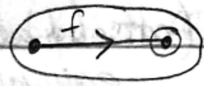
(these are binary & nullary composition)
comp. of 2 or 0 things

satisfying:

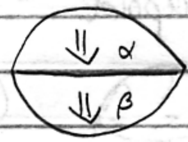
$\to \bullet \to \bullet \to$ associativity $(fg)h = f(gh)$

$\bullet \xrightarrow{f} \bullet$ $\quad 1_x f = f$ left unit law

chop up into 0-dim'l cell, & $\to$



$f 1_y = f$ right unit law



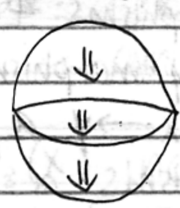
vertical composition of 2-morphisms

α, β give $\alpha\beta$

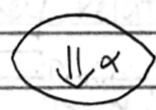
$x \xrightarrow{f} y$ gives identity 2-morphisms

f gives $1_f : f \Rightarrow f$

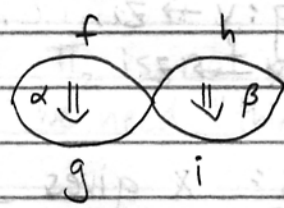
satisfying



associativity



left/right unit laws for vertical comp.

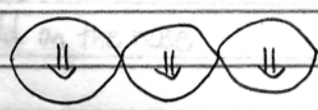


horizontal comp.

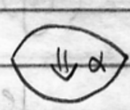
$\alpha : f \Rightarrow g$, $\beta : h \Rightarrow i$

give $\alpha \cdot \beta : fg \Rightarrow hi$

satisfying



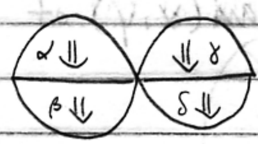
associativity



left/right unit laws

(Note – we can prove everything about whiskering from what we've defined.)

AND



interchange law for vert/horiz composition

$$(\alpha\beta)\cdot(\gamma\delta) = (\alpha\cdot\gamma)(\beta\cdot\delta)$$

But we can write this defn more simply:

If we categorify a category, we should get a 2-category.

Defn

A category is a set C_0 of objects,

$\forall\, x,y \in C_0$, a set $\hom(x,y)$ of morphisms,

$\forall\, x \in C_0$ an identity element $1_x \in \hom(x,x)$

$\forall\, x,y,z$ a composition function

$$\circ : \hom(x,y) \times \hom(y,z) \longrightarrow \hom(x,z)$$

st left/right unit laws & assoc. hold.

Now – we can categorify this object to get a 2-category.

"Categorified defn of category"

we get:

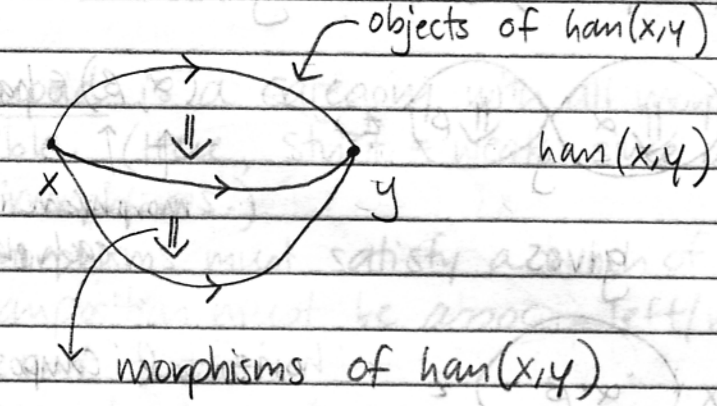
- A set C_0 of objects
- $\forall$ objects $x, y \in C_0$ a category $\hom(x,y)$ of morphisms
- $\forall$ objects $x \in C_0$ an identity object $1_x \in \hom(x,x)$ (element becomes an object)
- $\forall\, x,y,z$ a composition functor $\circ : \hom(x,y) \times \hom(y,z) \longrightarrow \hom(x,z)$

st left/right unit laws and assoc. hold on the nose (strict).

This above defn is what we call a 2-category.

(had we categorified everything correctly we'd get a double-category)

let's show this is the same as our defn of 2-category on prev pgs.



• these look like morphisms & 2-morphisms

Since hom(x,y) is a category, we can compose the double arrows, so we get vertical composition.

Now hom(x,y) × hom(y,z) is a product of categories
an object in here is a pair of objects, one each in hom(x,y), hom(y,z)
same for morphisms

On objects, $\circ : hom(x,y) \times hom(y,z) \longrightarrow hom(x,z)$

$x \xrightarrow{f} y \xrightarrow{g} z$ $\quad (f,g) \in hom(x,y) \times hom(y,z)$

gives

$x \xrightarrow{fg} z$ gives normal comp.

on morphisms, composition gives

$$\begin{tikzcd} x \arrow[r, bend left=50] \arrow[r, bend right=50] \arrow[r, phantom, "\Downarrow\alpha"] & y \arrow[r, bend left=50] \arrow[r, bend right=50] \arrow[r, phantom, "\Downarrow\beta"] & z \end{tikzcd}$$

$(\alpha, \beta) \in \hom(x,y) \times \hom(y,z)$

morphism in each place.

gives

$$\begin{tikzcd} x \arrow[r, bend left=50] \arrow[r, bend right=50] \arrow[r, phantom, "\alpha \cdot \beta"] & z \end{tikzcd}$$

horizontal composition

We need to check the laws...

most interesting: the interchange law

$$\begin{tikzcd} \bullet \arrow[r, bend left=60] \arrow[r] \arrow[r, bend right=60] \arrow[r, phantom, shift left=2ex, "\Downarrow"] \arrow[r, phantom, shift right=2ex, "\Downarrow"] & \bullet \arrow[r, bend left=60] \arrow[r] \arrow[r, bend right=60] \arrow[r, phantom, shift left=2ex, "\Downarrow"] \arrow[r, phantom, shift right=2ex, "\Downarrow"] & \bullet \end{tikzcd}$$

$\circ$ a functor means we can do in either order vert. comp then horiz. comp or vice versa

- vertical comp is from fact that $\hom(x,y)$ a cat.
- The fact that $\circ$ is a functor gives the interchange law!

*check this