

Spaces,  $\mathcal{S}$  will replace Set.  $\mathcal{S}$  is cartesian closed, so can enrich.  $\text{Cat}_0 := \mathcal{S}$ ,  $\text{Cat}_{n+1} := \text{Cat}_n$ -enriched categories

Recall:  $\text{Cat}_n \equiv (\infty, n)\text{Cat}$ .

$$\text{Cat}_\infty := \text{colim}_{n \rightarrow \infty}^{\text{Pr}^L} \left( \mathcal{S} \hookrightarrow \text{Cat}_1 \hookrightarrow \dots \hookrightarrow \text{Cat}_n \begin{array}{c} \xrightarrow{\perp} \\ \xrightarrow{\perp} \end{array} \text{Cat}_{n+1} \hookrightarrow \dots \right)$$

This is a (1-) category whose objects are

given by sequences  $(C_0, C_1, \dots)$  w/  $C_i \in \text{Cat}_i$  and  $C_n \cong C_{n+1}$ .

- Want: "oriented versions" of homotopy theory.

E.g.  $\mathcal{S}_* =$  pointed spaces.  $S'$  the circle &  $\forall X \in \mathcal{S}$ ,

$$\Omega_b X := \text{Maps}_{\mathcal{S}_*}(S', X)$$

Alternatively: homotopy pullback

$$\begin{array}{ccc} \Omega_b X & \longrightarrow & \bullet \\ \downarrow \Gamma_h & & \downarrow \\ \bullet & \longrightarrow & X \end{array}$$

or strict pullback

$$\begin{array}{ccc} \Omega_b X & \longrightarrow & PX \\ \downarrow \Gamma & & \downarrow \\ \bullet & \longrightarrow & X \end{array}$$

Write  $S' = B\mathbb{Z}$ . Can we "directify" by replacing  $\mathbb{Z}$  w/  $\mathbb{N}$ ?

i.e. for  $\mathcal{C} \in \text{Cat}_{\infty,*}$ , should we define  $\Omega_b \mathcal{C}$  as  $\text{Maps}_{\text{Cat}_{\infty,*}}(B\mathbb{N}, \mathcal{C})$ .

These give endo's at base pts like we want, but  $\text{Cat}_{\infty,*}$  is

an  $(\infty, 1)$  category, so  $\text{Maps}_{\text{Cat}_{\infty,*}}(B\mathbb{N}, \mathcal{C}) \in \mathcal{S}$  ~~must~~ only has

invertible higher cells, so not really " $\mathcal{C}(\text{pt}, \text{pt})$ ".

$$\text{Maps}_{\text{Cat}_\infty}(\mathcal{C}, \mathcal{D}) = \begin{cases} \text{obj} = \text{functors } \mathcal{C} \rightarrow \mathcal{D} \\ \text{mor} = \text{pseudonatural transformation} \\ \text{etc.} \end{cases}$$

Solution; replace pseudo w/ (op)lax:

$\Omega \mathcal{C} := \text{Maps}_{\text{Cat}_\infty}^{\text{lax}}(\mathbb{B}N, \mathcal{C})$ , so we'll need some kind of

oriented / (op)lax pullback;  $\Omega \mathcal{C} \longrightarrow \cdot$   
 $\downarrow \Gamma \nearrow \downarrow$   
 $\cdot \longrightarrow \mathcal{C}$

In non-pointed setting,  $\mathcal{C}(x, y) \longrightarrow \cdot$   
 $x \downarrow \Gamma \nearrow \downarrow y$   
 $\cdot \longrightarrow \mathcal{C}$

"comma pullback"  $\{Fx \rightarrow Gy\} \rightarrow \mathcal{C}$   
 $F \downarrow \Gamma \nearrow \downarrow G$   
 $\mathcal{D} \longrightarrow \mathcal{E}$

**Prop<sup>1</sup>**:  $\text{Maps}_{\text{Cat}_\infty}^{\text{lax}}(\mathcal{C}, \mathcal{D})$  exists; i.e. there's a monoidal structure

$\boxtimes$  on  $\text{Cat}_\infty$  and it has an adjoint:

$$\text{Cat}_\infty(\mathcal{C} \boxtimes \mathcal{D}, \mathcal{E}) \cong \text{Cat}_\infty(\mathcal{C}, \text{Cat}_\infty^{\text{lax}}(\mathcal{D}, \mathcal{E}))$$

$\text{Cat}_\infty$  has ( $\geq$ ) 2 closed monoidal structures,  $\times$  &  $\boxtimes$ , thus

it is self-enriched in ( $\geq$ ) 2 ways.  $\Omega: \text{Cat}_\infty \rightarrow \text{Cat}_\infty$  is

not a functor of  $(\text{Cat}_\infty, \times)$ -categories, but is a functor

of  $(\text{Cat}_\infty, \boxtimes)$ -categories.  $\boxtimes: \text{Cat}_\infty \times \text{Cat}_\infty \rightarrow \text{Cat}_\infty$  is called

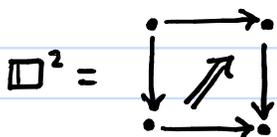
the **Gray product**.  $(\text{Cat}_\infty, \boxtimes)$ -categories are called

**oriented categories**.  $\exists \Sigma \dashv \Omega$  w/  $\Sigma = \mathbb{B}\text{Free}_\otimes(-)$

More on  $\boxtimes$ :

$$\boxtimes^0 = *$$

$$\boxtimes^1 = \cdot \longrightarrow \cdot$$



etc.

$$\begin{aligned} \boxtimes^1 \boxtimes \boxtimes^1 &\text{ defined by: } \text{Cat}_\infty(\boxtimes^1 \boxtimes \boxtimes^1, \mathcal{C}) \\ &\cong \text{Cat}_\infty(\boxtimes^1, \text{Cat}_\infty^{\text{lax}}(\boxtimes^1, \mathcal{C})) \end{aligned}$$

Since  $\boxtimes^1 = \cdot \longrightarrow \cdot$  &  $\text{Cat}_\infty^{\text{lax}}(\boxtimes^1, \mathcal{C})$  has objects morphisms in  $\mathcal{C}$  & morphisms lax transformations. Thus  $\boxtimes^1 \boxtimes \boxtimes^1$  picks out

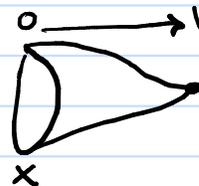
squares in  $\mathcal{C}$  of the form  $\begin{array}{ccc} & \xrightarrow{?} & \\ \downarrow & \nearrow & \downarrow \\ & \xrightarrow{?} & \end{array}$ , so  $\boxtimes^1 \boxtimes \boxtimes^1 = \boxtimes^2$ .

More generally,  $\boxtimes^n \boxtimes \boxtimes^m = \boxtimes^{n+m}$

## Joins

For spaces; start w/ cone:  $X \hookrightarrow X * [0, 1]$

$$\begin{array}{ccc} \downarrow & \lrcorner & \downarrow \\ \cdot & \longrightarrow & \mathcal{C}X \end{array}$$



Join is two cones glued at tips:

$$\begin{array}{ccc} \cdot & \longrightarrow & \mathcal{C}X \\ \downarrow & \lrcorner & \downarrow \\ \mathcal{C}Y & \longrightarrow & X * Y \end{array}$$



For categories; lax cone  $X \hookrightarrow X \boxtimes \boxtimes^1$

$$\begin{array}{ccc} \downarrow & \cong & \downarrow \\ \cdot & \longrightarrow & \mathcal{C}X \end{array}$$

$$\begin{array}{ccc}
 \Omega \mathcal{C} & \longrightarrow & \cdot \\
 \downarrow \Gamma & \nearrow & \downarrow \\
 \cdot & \longrightarrow & \mathcal{C}
 \end{array}
 \quad \text{or} \quad
 \begin{array}{ccc}
 \Omega \mathcal{C} & \longrightarrow & \text{Cat}_{\infty}^{\text{Lax}}(\boxtimes', \mathcal{C}) \\
 \downarrow \Gamma_{\text{strict!}} & & \downarrow \\
 \cdot & \longrightarrow & \mathcal{C}
 \end{array}$$

Similarly

$$\begin{array}{ccc}
 \mathcal{C} & \longrightarrow & \cdot \\
 \downarrow & \nearrow & \downarrow \\
 \cdot & \longrightarrow & \Sigma \mathcal{C}
 \end{array}
 \quad \text{or} \quad
 \begin{array}{ccc}
 \mathcal{C} & \longrightarrow & \cdot \\
 \downarrow & \text{strict!} & \downarrow \\
 \mathcal{C} \boxtimes \boxtimes' & \longrightarrow & \Sigma \mathcal{C}
 \end{array}$$

**Remark:**  $\boxtimes$  is not symmetric.  $\boxtimes^{\text{rev}}$  has inner hom given by  $\text{Cat}_{\infty}^{\text{oplax}}(-, -)$

$\rightsquigarrow$  Bi-enrichment; enrich over  $\boxtimes$  &  $\boxtimes^{\text{rev}}$  at same time.