Unifying graded and parameterised monads

Dominic Orchard
University of Kent

Philip Wadler
The University of Edinburgh

Harley Eades III
Augusta University

MSFP 2020

The humble monad

$$T : \mathbb{C} \to \mathbb{C}$$

`join :: m m a -> m a`

$$\mu : T \circ T \to T$$

multiplication

$$\eta : \mathrm{Id} \to T$$

unit

`return :: a -> m a`

+ associativity and unitality axioms

Graded monads

$$G : \mathbb{E} \to [\mathbb{C}, \mathbb{C}]$$ Functor

$$(\mathbb{E}, \bullet, I)$$ (Discrete) monoidal category

$$\mu_{x,y} : Gx \circ Gy \to G(x \bullet y)$$

multiplication

$$\eta : \mathrm{Id} \to G\,I$$

unit

+ associativity and unitality axioms

[Wadler&Thiemman'03 - Marriage of effects and monads]

[Katsumata'14 - Parametric effect monads and semantics of effect systems]

Graded monads for type-based effect analysis

monadic metalanguage

$$\frac{\Gamma \vdash e : MA \quad \Gamma, x : A \vdash e' : MB}{\Gamma \vdash \textbf{do } x \leftarrow e; e' : MB}$$

$$\frac{\Gamma \vdash e : A}{\Gamma \vdash \textbf{return } e : MA}$$

Graded monads for type-based effect analysis

graded
monadic metalanguage

$$\frac{\Gamma \vdash e : G\,x\,A \quad \Gamma, x : A \vdash e' : G\,y\,B}{\Gamma \vdash \textbf{do}\ x \leftarrow e; e' : G\,(x \bullet y)\,B}$$

$$\frac{\Gamma \vdash e : A}{\Gamma \vdash \textbf{return}\ e : G\,I\,A}$$

... for refining semantics

Humble state

$$\text{State}\,A = \text{Store}(\mathscr{L}) \to A \times \text{Store}(\mathscr{L})$$

$$\text{get} : (l : A \in \mathscr{L}) \to \text{State}\ A$$

$$\text{put} : (l : A \in \mathscr{L}) \to A \to \text{State}\ 1$$

vs. graded

$$\text{State}\,x\,A = \text{Store}(\text{reads}(x)) \to A \times \text{Store}(\text{writes}(x))$$

$$\text{get} : (l : A \in \mathscr{L}) \to \text{State}\ \{r(l)\}\ A$$

$$\text{put} : (l : A \in \mathscr{L}) \to A \to \text{State}\ \{w(l)\}\ 1$$

Graded monads

$$G : \mathbb{E} \to [\mathbb{C}, \mathbb{C}]$$ Functor

$$(\mathbb{E}, \bullet, I)$$ (Discrete) monoidal category

$$G\,x \circ G\,y \xrightarrow{\mu_{x,y}} G\,(x \bullet y)$$

multiplication

$$\mathrm{Id} \xrightarrow{\eta} G\,I$$

unit

+ associativity and unitality axioms

(unordered) Graded monads

$G : \mathbb{E} \to [\mathbb{C}, \mathbb{C}]$ Functor

$(\mathbb{E}, \bullet, I)$ (Discrete) monoidal category

$$Gx \circ Gy \xrightarrow{\mu_{x,y}} G(x \bullet y)$$

multiplication

$$\mathrm{Id} \xrightarrow{\eta} GI$$

unit

+ associativity and unitality axioms

Graded monads

$G : \mathbb{E} \to [\mathbb{C}, \mathbb{C}]$ Functor

$(\mathbb{E}, \bullet, I, \leq)$ Strict monoidal category

$$G\,x \circ G\,y \xrightarrow{\mu_{x,y}} G\,(x \bullet y)$$

multiplication

$$\mathrm{Id} \xrightarrow{\eta} G\,I$$

unit

$$G\,x \xrightarrow{G(h : x \leq y)} G\,y$$

approximation

+ associativity and unitality axioms

+ monotonicity

unordered

graded monads

Parametrised monads

$$P : \mathbb{I}^{\mathsf{op}} \times \mathbb{I} \to [\mathbb{C}, \mathbb{C}]$$ Functor

$$P(i,j) \circ P(j,k) \xrightarrow{\mu_{i,j,k}} P(i,k)$$

multiplication

$$\mathsf{Id} \xrightarrow{\eta_i} P(i,i)$$

unit

+ associativity and unitality axioms

[Atkey'06/'09 - Parameterised notions of computation]

Parametrised monads

$$P : \mathbb{I}^{\mathrm{op}} \times \mathbb{I} \to [\mathbb{C}, \mathbb{C}]$$ Functor

$$P(i,j) \circ P(j,k) \xrightarrow{\mu_{i,j,k}} P(i,k)$$

multiplication

$$\mathrm{Id} \xrightarrow{\eta_i} P(i,i)$$

unit

+ associativity and unitality axioms

$$\frac{A \xrightarrow{f} P(i,j)\,B \qquad B \xrightarrow{g} P(j,k)\,C}{A \xrightarrow[\mu_{i,j,k,C} \circ Pg \circ f]{} P(i,k)\,C}$$

cf Floyd-Hoare logic

$$\frac{\{i\}\,C\,\{j\} \qquad \{j\}\,C'\,\{k\}}{\{i\}\,C; C'\,\{k\}}$$

[Atkey'06/'09 - Parameterised notions of computation]

Parametrised monads

$$P : \mathbb{I}^{op} \times \mathbb{I} \to [\mathbb{C}, \mathbb{C}]$$ Functor

$$\mu_{i,j,k} : P(i,j) \circ P(j,k) \to P(i,k)$$

multiplication

$$\eta_i : \mathrm{Id} \to P(i,i)$$

unit

$$P(f : i' \to i, g : j \to j') : P(i,j) \to P(i',j')$$

approximation

+ associativity and unitality axioms

+ dinaturality axioms

[Atkey'06/'09 - Parameterised notions of computation]

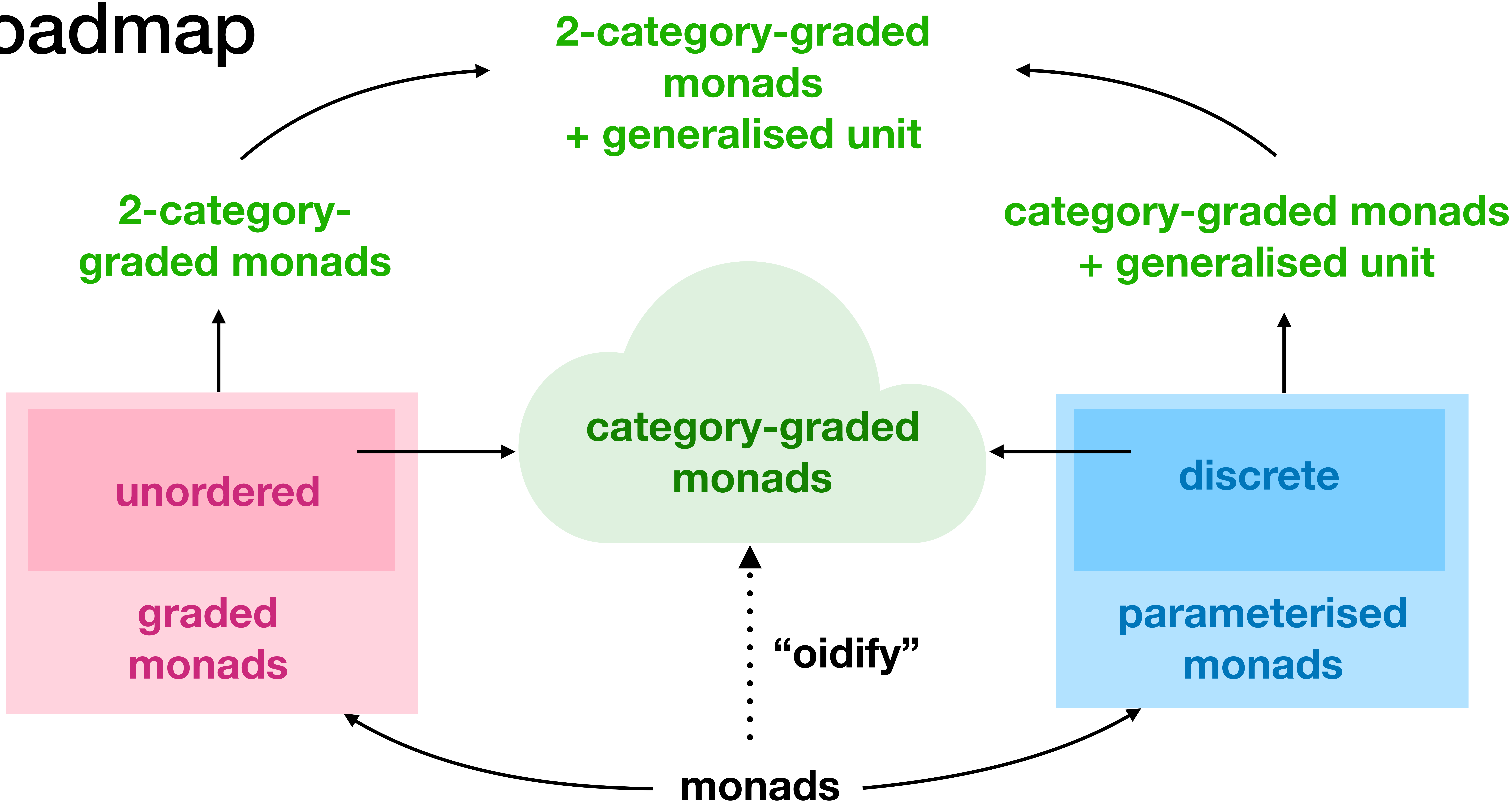
G and P: side-by-side

$$\begin{array}{c} G\,x \circ G\,y \\ \mu_{x,y} \big\downarrow \\ G\,(x \bullet y) \end{array} \qquad \begin{array}{c} P(i,j) \circ P(j,k) \\ \mu_{i,j,k} \big\downarrow \\ P\,(i,k) \end{array}$$

$$\begin{array}{c} \mathtt{Id} \\ \eta \big\downarrow \\ G\,I \end{array} \qquad \begin{array}{c} \mathtt{Id} \\ \eta_i \big\downarrow \\ P\,(i,i) \end{array}$$

Can we unify their definitions?

Roadmap



2-category-graded
monads
+ generalised unit
2-category-
graded monads
category-graded monads
+ generalised unit
unordered
graded
monads
category-graded
monads
discrete
parameterised
monads
"oidify"
monads

“Oidification”

nLab

1. concept shown to be equivalent to a single-object category

2. generalise that to a category with more than one object/morphism

https://ncatlab.org/nlab/show/horizontal+categorification

Monads are lax functors

(Benabou 1967)

One object $\star$

One morphism

$$id : \star \to \star$$

$$T : 1 \to \text{Endo}(\mathbb{C})$$

One object $\mathbb{C}$

Morphisms are endofunctors

2-morphisms are nat. trans.

Recall: functor axioms

$F : \mathbb{C} \to \mathbb{D}$

$$id = F\,id$$

$$Fg \circ Ff = F(g \circ f)$$

Lax functor axioms

$$id \Rightarrow F\,id$$

$$Fg \circ Ff \Rightarrow F(g \circ f)$$

+ associativity/unitality

$\mathbb{D}$ is a 2-category

Here natural transformations

$$\eta : \quad id \Rightarrow T\,id$$

$$\mu : T id \circ T id \Rightarrow T id$$

Oidifying a monad

Monad | **"Category-graded monad"**

$$1 \xrightarrow{T} \mathrm{Endo}(\mathbb{C})$$

oidification / horizontal categorification $\cdots\cdots\to$

$$\mathscr{F}^{\mathrm{op}} \xrightarrow{T} \mathrm{Endo}(\mathbb{C})$$

(Benabou 1967)

Lax functor

$$\eta : \mathrm{Id} \Rightarrow T\,id \qquad \eta_x : \mathrm{Id} \Rightarrow T\,id_x$$

$$\mu : T\,id \circ T\,id \Rightarrow T\,id \qquad \mu_{f,g} : Tf \circ Tg \Rightarrow T(g \circ f)$$

1. Unordered graded monads are category-graded monads

"Category-graded monad"

$$\mathscr{F}^{\mathrm{op}} \xrightarrow{T} \mathrm{Endo}(\mathbb{C})$$

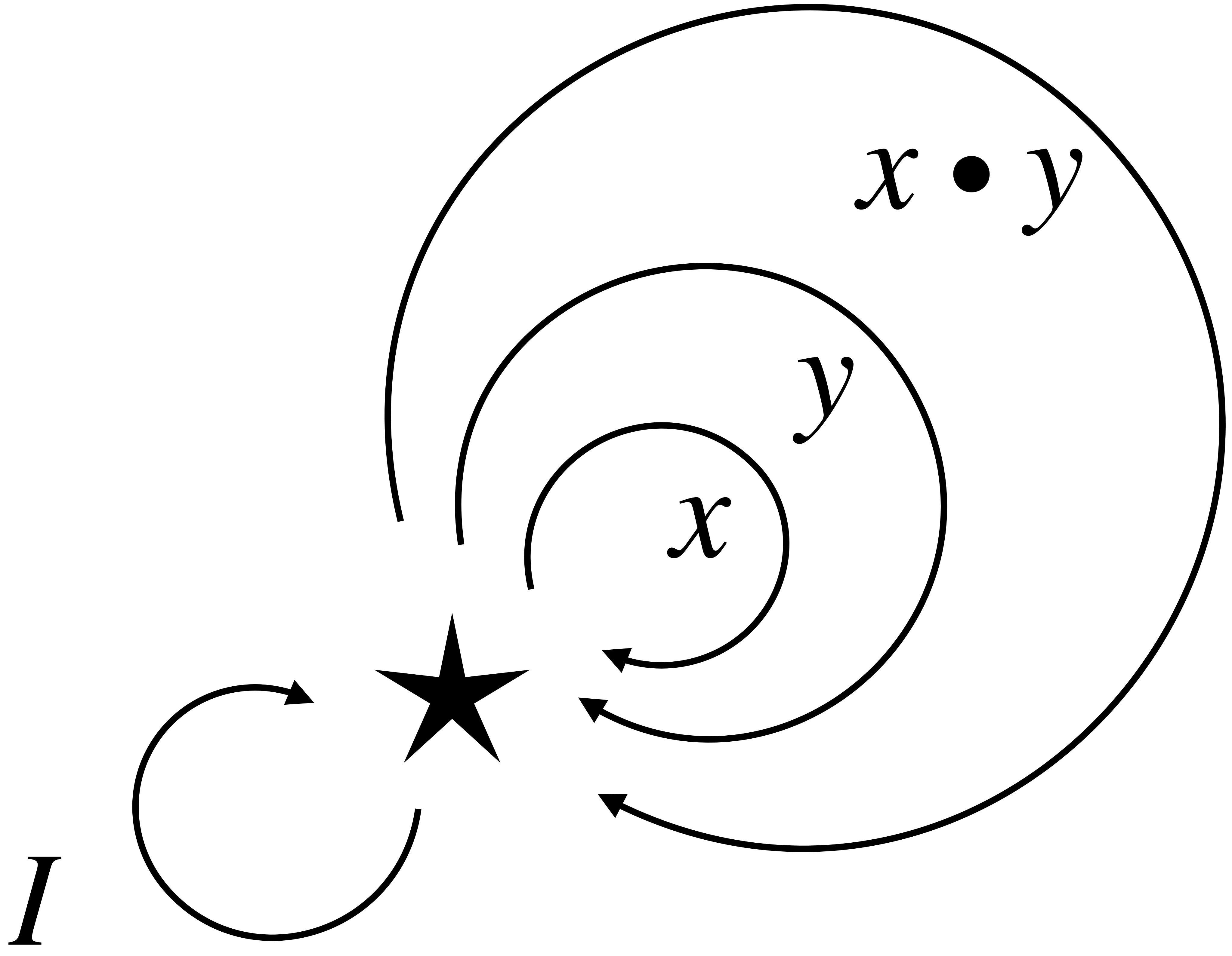
$$\eta_x : \mathrm{Id} \Rightarrow T\, id_x$$

$$\mu_{f,g} : Tf \circ Tg \Rightarrow T(g \circ f)$$

Monoid-graded monad G

$$(\mathbb{E}, \bullet, I) \xrightarrow{G} [\mathbb{C}, \mathbb{C}]$$

<u>Monoids</u> are <u>one-object categories</u> are <u>discrete monoidal categories</u>



1. Unordered graded monads are category-graded monads

Category-graded monad

$$\mathscr{F}^{\mathrm{op}} \xrightarrow{\;T\;} \mathrm{Endo}(\mathbb{C})$$

$\supseteq$

Monoid-graded monad G

$$1_{(\mathbb{E}^{op},\bullet,I)} \equiv (\mathbb{E},\bullet,I)$$

$$\downarrow \qquad\qquad \downarrow G$$

$$\mathrm{Endo}(\mathbb{C}) \equiv [\mathbb{C},\mathbb{C}]$$

$\eta_x : \mathrm{Id} \Rightarrow T\, id_x$

$\mu_{f,g} : Tf \circ Tg \Rightarrow T(g \circ f)$

$\eta_\star : \mathrm{Id} \Rightarrow G\, I$

$\mu_{x,y} : G\, x \circ G\, y \Rightarrow G\,(x \bullet y)$

2. Graded monads are 2-category-graded monads

2-category-graded monad

Category-graded monad

$$T : \mathscr{F}^{\mathrm{op}} \to \mathrm{Endo}(\mathbb{C})$$

$$\eta_x : \mathrm{Id} \Rightarrow T\, id_x$$

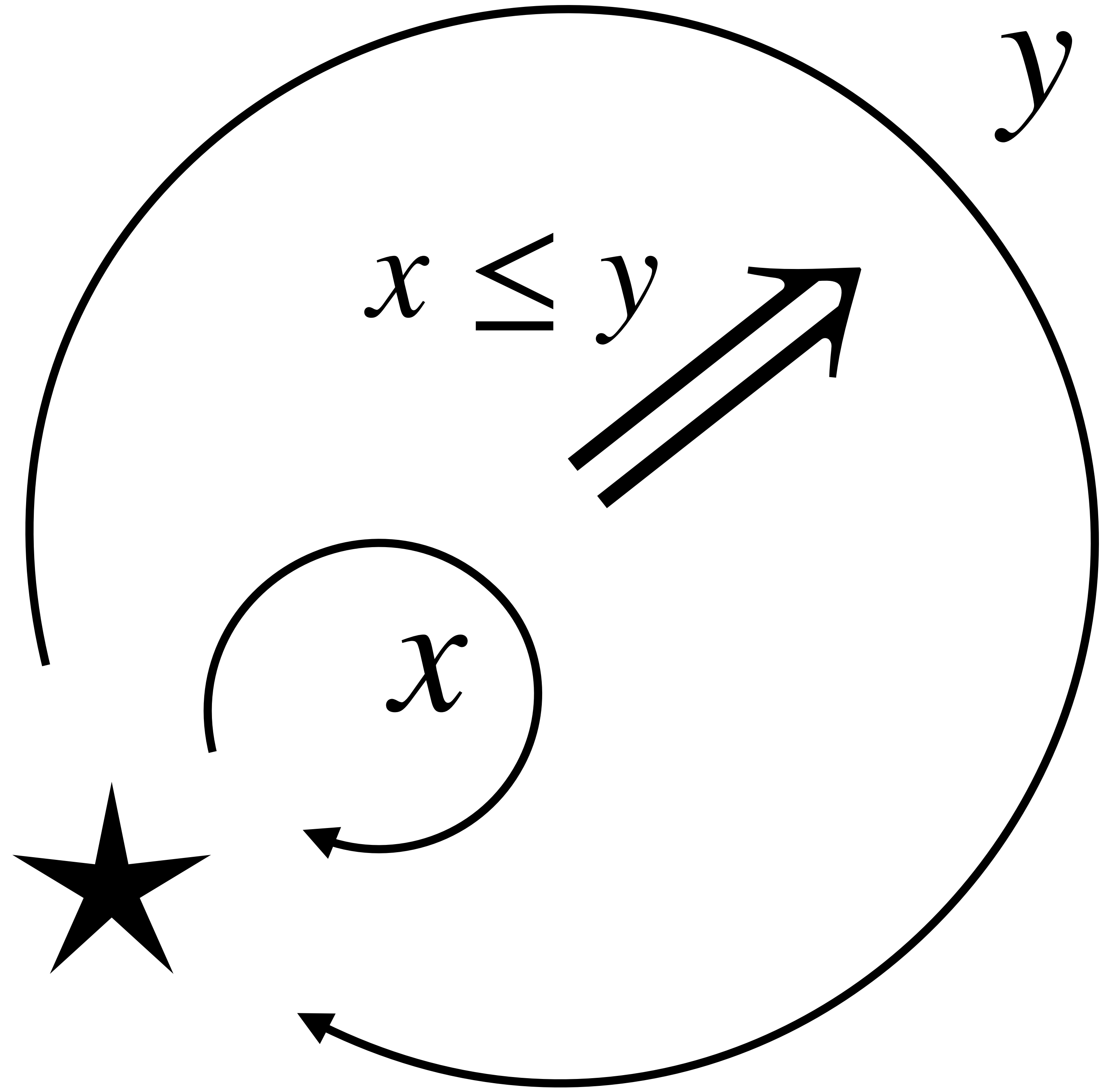
$$\mu_{f,g} : Tf \circ Tg \Rightarrow T(g \circ f)$$

+ 2-morphism mapping

$$T(\mathbf{h} : g \Rightarrow f) : T\,g \Rightarrow T\,f$$

(i.e., $\mathscr{F}$ is a 2-category)

Pomonoids are one object 2-categories are monoidal categories



2. Graded monads are 2-category-graded monads

2-category-graded monad

$$T : \mathcal{F}^{\mathsf{op}} \to \mathsf{Endo}(\mathbb{C})$$

$$\eta_x : \mathsf{Id} \Rightarrow T\, id_x$$

$$\mu_{f,g} : Tf \circ Tg \Rightarrow T(g \circ f)$$

$$T(\mathbf{h} : g \Rightarrow f) : T\, g \Rightarrow T\, f$$

2-morphism mapping

$\supseteq$

(Ordered) Graded monad

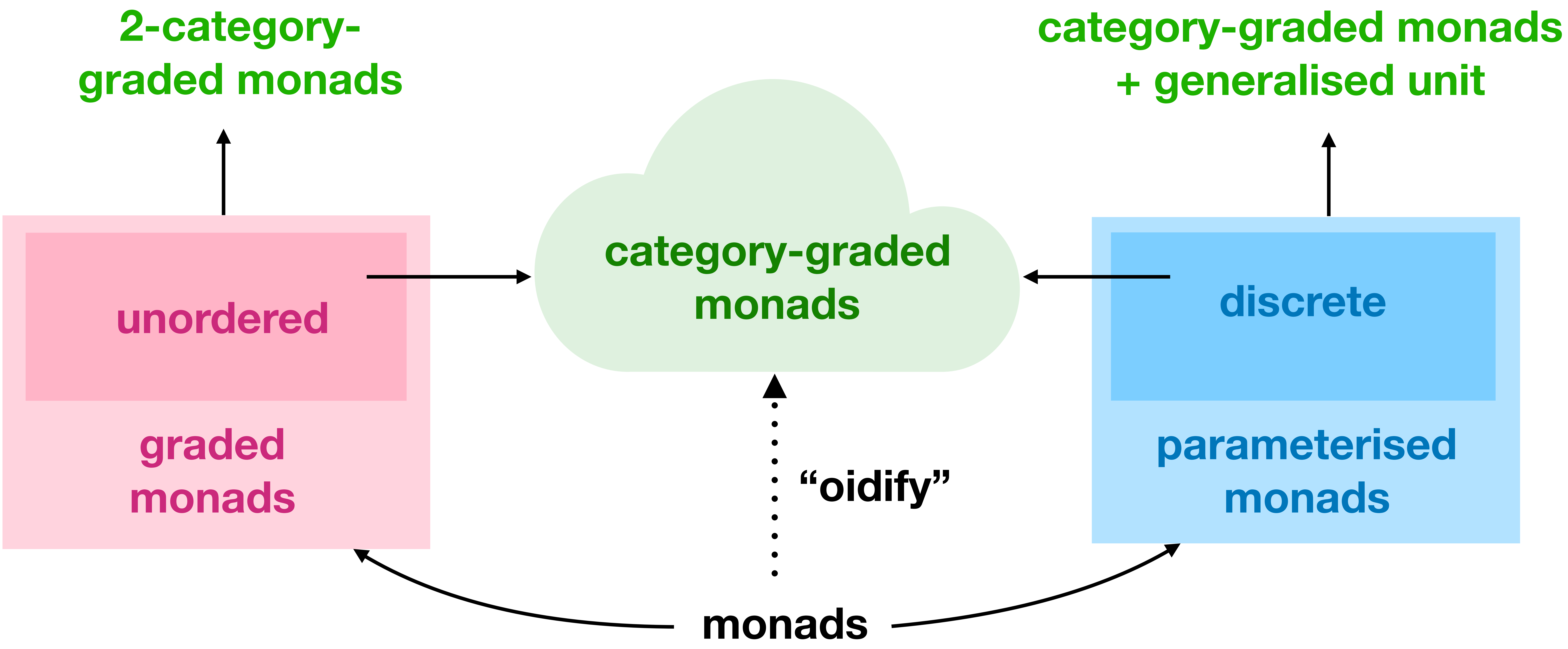
$$G : 1_{(\mathbb{E}^{op}, \bullet, I, \leq)} \to \mathsf{Endo}(\mathbb{C})$$

$$\eta_\star : \mathsf{Id} \Rightarrow G\, I$$

$$\mu_{x,y} : G\, x \circ G\, y \Rightarrow G\,(x \bullet y)$$

$$G(h : x \leq y) : G\, x \Rightarrow G\, y$$

Roadmap


2-category-graded monads
category-graded monads + generalised unit
unordered
graded monads
category-graded monads
discrete
parameterised monads
"oidify"
monads

3. **Discrete** parameterised monads are category-graded monads

$P : \mathbb{J}^{\mathsf{op}} \times \mathbb{J} \to [\mathbb{C}, \mathbb{C}]$ where $\mathbb{J}$ has only identity morphisms

Define the category of $\mathbb{J}$-"dominoes"

$$\nabla(\mathbb{J})_1 = |\mathbb{J}| \times |\mathbb{J}|$$

(e.g., composition $(j, k) \circ (i, j) = (i, k)$)

Define a category graded monad

$$T : \nabla(\mathbb{J}) \to \mathsf{Endo}(\mathbb{C})$$

with $T(i, j) = P(i, j)$

$$\eta_i : \mathsf{Id} \Rightarrow T(i, i) = \eta_i^P$$

$$\mu_{(i,j),(j,k)} : T(i, j) \circ T(j, k) \Rightarrow T(i, k) = \mu_{i,j,k}^P$$

Parameterised monads have some extra structure

$$P\,(i, j)$$

$$\Big\downarrow\; P(f : i' \to i, g : j \to j')$$

$$P\,(i', j')$$

morphism mapping (approximation) + dinaturality axioms

Generalised units

arises from lax natural transformations (Street, 1972)

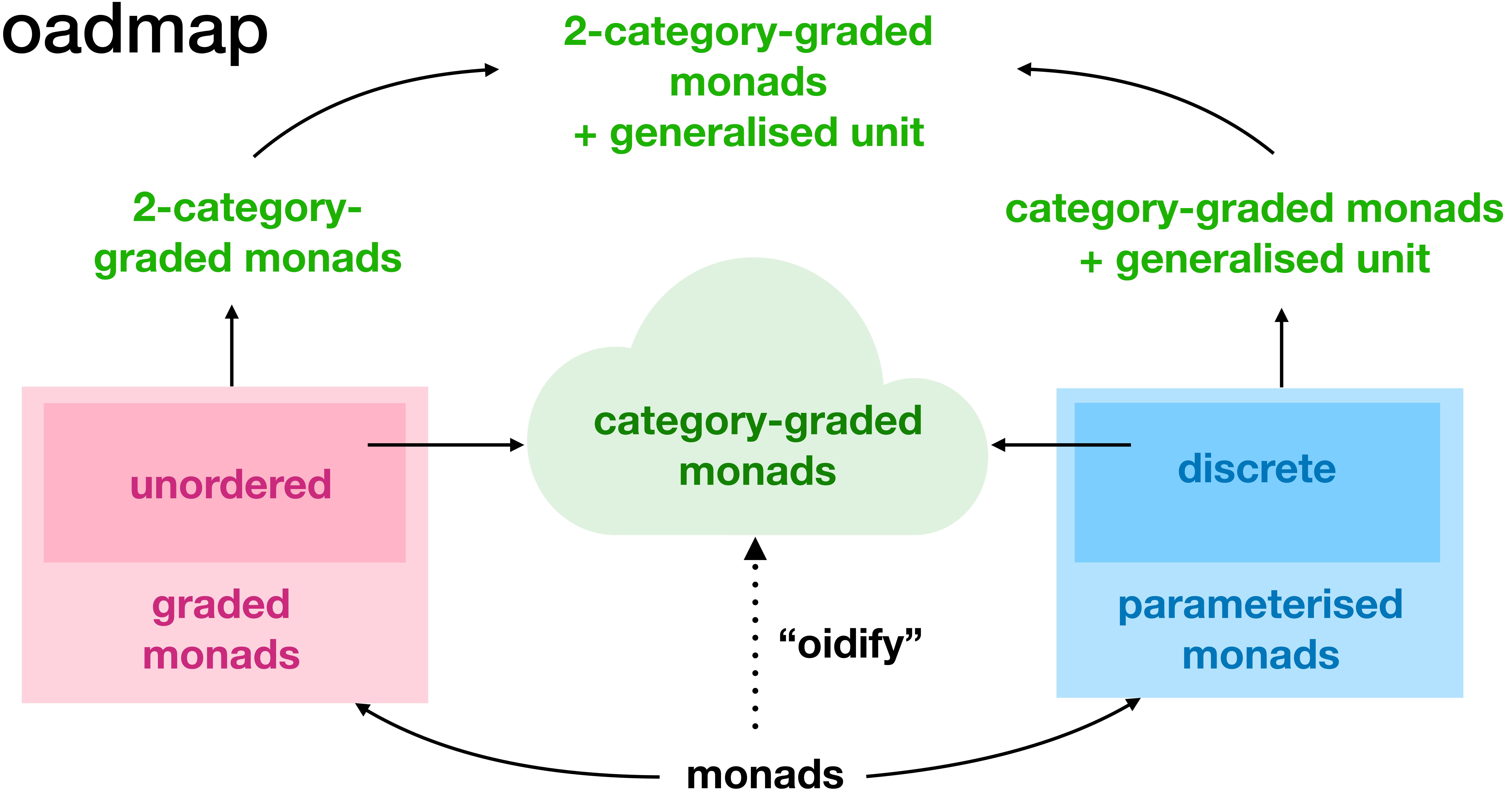
Wide sub-category $\mathbb{S} \subseteq \mathscr{F}$

Family of morphisms $\hat{\eta}_{f:X \to Y \in \mathbb{S}} : \mathtt{Id} \to Tf$

4. Parameterised monads are category-graded monads + $\hat{\eta}$

Paper shows details

Roadmap


2-category-graded monads + generalised unit
2-category-graded monads
category-graded monads + generalised unit
category-graded monads
unordered
graded monads
discrete
parameterised monads
"oidify"
monads

Example

$\mathsf{ConcSt} : \mathscr{F}^{\mathsf{op}} \to \mathsf{Endo}(\mathbb{C})$

$|\mathscr{F}| = \{\textbf{free}, \textbf{critical}\}$

$\textbf{lock} : \textbf{free} \to \textbf{critical}$

$\textbf{unlock} : \textbf{critical} \to \textbf{free}$

$\textbf{get}, \textbf{put} : \textbf{critical} \to \textbf{critical}$

$get : \mathsf{ConcSt}\,\textbf{get}\ S$

$put : S \to \mathsf{ConcSt}\,\textbf{put}\ 1$

$lock : \mathsf{ConcSt}\,\textbf{lock}\ 1$

$unlock : \mathsf{ConcSt}\,\textbf{unlock}\ 1$

$\textbf{spawn} : (\forall f.\, \mathsf{ConcSt}\,(f : \mathsf{free} \to \mathsf{free})\ 1) \to \mathsf{ConcSt}\ f\ 1$

Conclusions

Category-graded monad

- Shows us where graded & parameterised overlap
- A more general structure that captures both aspects: tracing + restriction

$$\mathcal{F}^{\mathrm{op}} \xrightarrow{\;T\;} \mathrm{Endo}(\mathbb{C})$$

$$\eta_x : \mathrm{Id} \Rightarrow T\, id_x$$

$$\mu_{f,g} : Tf \circ Tg \Rightarrow T(g \circ f)$$

granule-project.github.io

Thank you!