Comparing semantic frameworks for dependently-sorted algebraic theories

Benedikt Ahrens, Peter LeFanu Lumsdaine, Paige Randall North

published in APLAS 2024
arXiv:2412.19946

Outline

Summary

Goal

Give precise relationship between different categorical structures used to interpret dependent types such as

- display map categories,
- comprehension categories,
- categories with families,
- contextual categories...

Motivation

1. Literature on existing notions is scattered and incomplete
2. New notions are frequently developed
3. Most of these are ‘the same’ or nice subcategories of others, but we wanted to write down the relationships clearly.

Methodology

Our approach:

1. Comprehension categories are the most general notion
2. Identify other notions as constituting certain subcategories of comprehension categories
3. Focus on the 2-categories of these notions.
4. But we also give a strict/1-categorical analysis

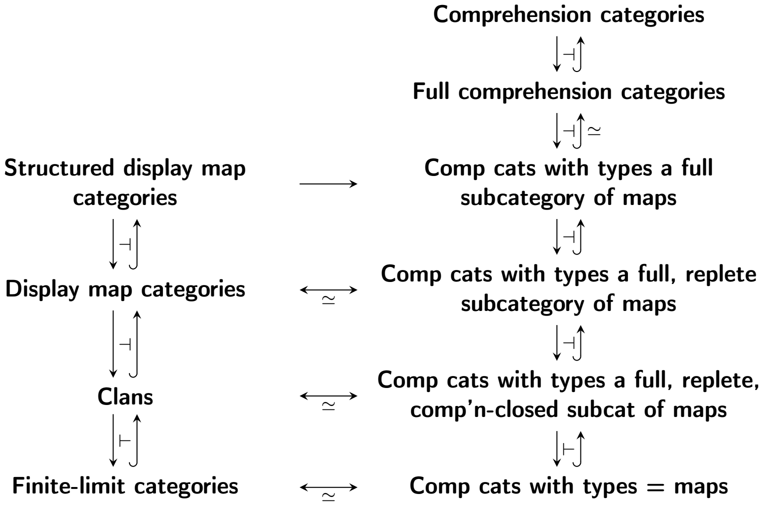
Why 2-categories?

There were previous analyses on the 1-categorical level, e.g., by Javier Blanco and Martin Hofmann.

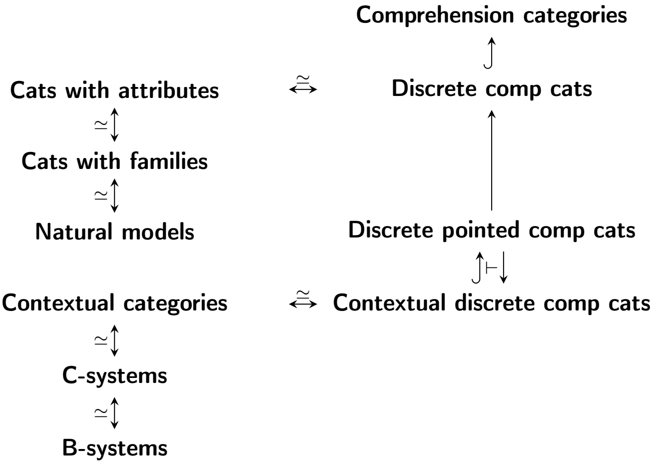
But we prefer a 2-categorical analysis:

- These are categories with structure, so they naturally form 2-categories.
- Want to have pseudo (weak) morphisms between these categories with structure, and need 2-morphisms to ‘control’ these.
  - These arise naturally: e.g., after applying Hofmann’s strictification

Frameworks with types as certain maps



Frameworks with types as primitive



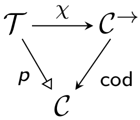
Outline

Comprehension categories

Definition

A **comprehension category** consists of

1. a category $\mathcal{C}$ (whose objects we call **contexts**);
2. a fibration $\mathcal{T} \xrightarrow{p} \mathcal{C}$ (of **types**); and
3. a functor $\mathcal{T} \xrightarrow{\chi} \mathcal{C}^{\to}$ (**comprehension**); such that
4. χ lies strictly over $\mathcal{C}$, in that $\mathrm{cod} \circ \chi = p$, and is cartesian, i.e. sends p-cartesian maps to pullback squares.



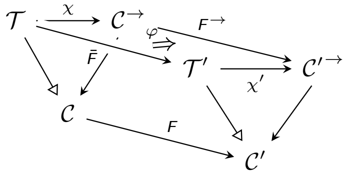
Pseudo maps of comprehension categories

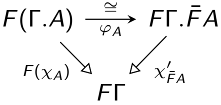
Definition

A **pseudo map** $(F, \bar{F}, \varphi) : (\mathcal{C}, \mathcal{T}, p, \chi) \longrightarrow (\mathcal{C}', \mathcal{T}', p', \chi')$ is:

1. a functor $F : \mathcal{C} \longrightarrow \mathcal{C}'$;
2. a functor $\bar{F} : \mathcal{T} \longrightarrow \mathcal{T}'$ lying (strictly) over F, and sending p-cartesian maps to p'-cartesian maps; and
3. a natural isomorphism $\varphi : \chi'\bar{F} \cong F^{\rightarrow}\chi$ lying (strictly) over the identity natural transformation on F

A **strict map** is a pseudo map where φ is the identity.





Summary: comprehension categories

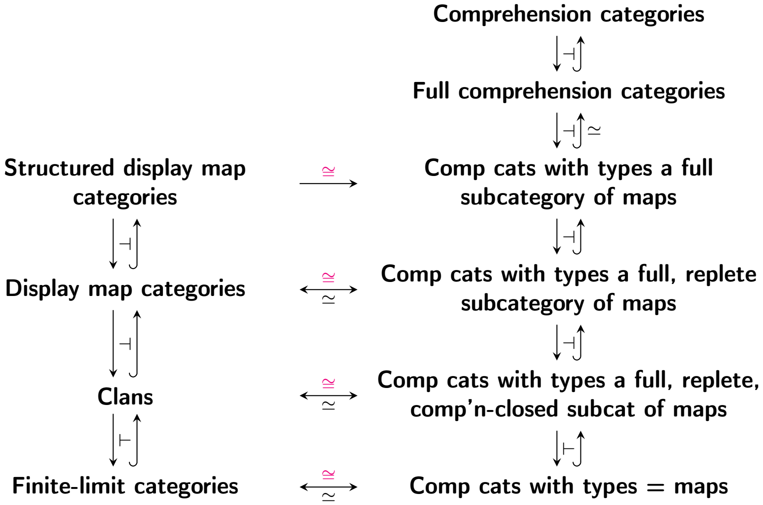
- Comprehension categories, pseudo maps (resp. strict maps), and transformations form a 2-category CompCat (resp. $\mathrm{CompCat}^{\mathrm{str2}}$).

Why (not) use comprehension categories?

\+ versatile

− morphisms of types don't mean anything

Frameworks with types as certain maps



NB! The right-hand sides of each $\cong$ is the 2-category with strict 1-morphisms.

Outline

Overview

Comprehension categories

Types as maps

Types as primitive

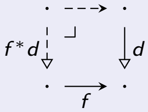
Conclusion

Display map categories

Definition

A **display map category** consists of

1. a category $\mathcal{C}$ together with
2. a replete (i.e. isomorphism-invariant) subclass $\mathcal{D} \subseteq \mathrm{mor}(\mathcal{C})$ of maps (called *display maps* and written $\longrightarrow\!\!\triangleright$)
3. such that display maps pull back along arbitrary maps:



Why (not) use display map categories?

+ easy to construct from mathematical objects
− relatively far removed from syntax

Structured display map categories

Definition

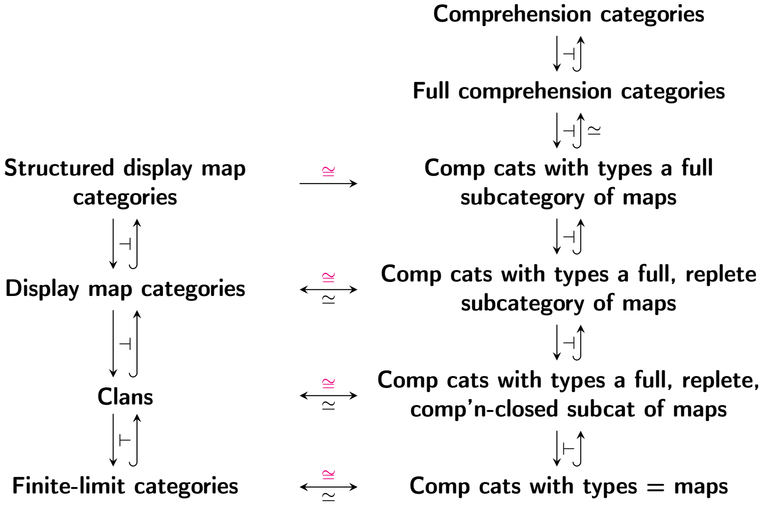
A **structured display map category** (sDMC) consists of

1. a category $\mathcal{C}$ together with
2. a subclass $\mathcal{D} \subseteq \mathrm{mor}(\mathcal{C})$ of maps (called *display maps* and written $\longrightarrow\!\!\triangleright$)
3. such that display maps pull back along arbitrary maps:

$$\begin{array}{ccc} \cdot & \dashrightarrow & \cdot \\ {\scriptstyle f^*d}\,\downarrow & \lrcorner & \downarrow\,{\scriptstyle d} \\ \cdot & \xrightarrow[f]{} & \cdot \end{array} \tag{1}$$

A **morphism** of sDMCs is a functor preserving display maps and pullbacks.

Frameworks with types as certain maps



NB! The right-hand sides of each ≅ is the 2-category with strict 1-morphisms.

Clans

Definition

A display map category is **rooted** if

1. $\mathcal{C}$ has a terminal object, and
2. all morphisms to the terminal object are composites of display maps and isomorphisms. (In the non-structured case, repleteness renders the isomorphisms redundant.)

Definition

A **clan** is a rooted display map category $(\mathcal{C}, \mathcal{D})$ where $\mathcal{D}$ is closed under composition and contains all identities.

Why (not) use clans?

+ easy to construct from mathematical objects
− relatively far removed from syntax

Finite-limit categories

Construction

Every finite-limit category $\mathcal{C}$ forms a clan $(\mathcal{C}, \mathrm{mor}(\mathcal{C}))$, so

$$\textbf{Finite-limit categories} \hookrightarrow \textbf{Clans}$$

This has a right adjoint given by the following:

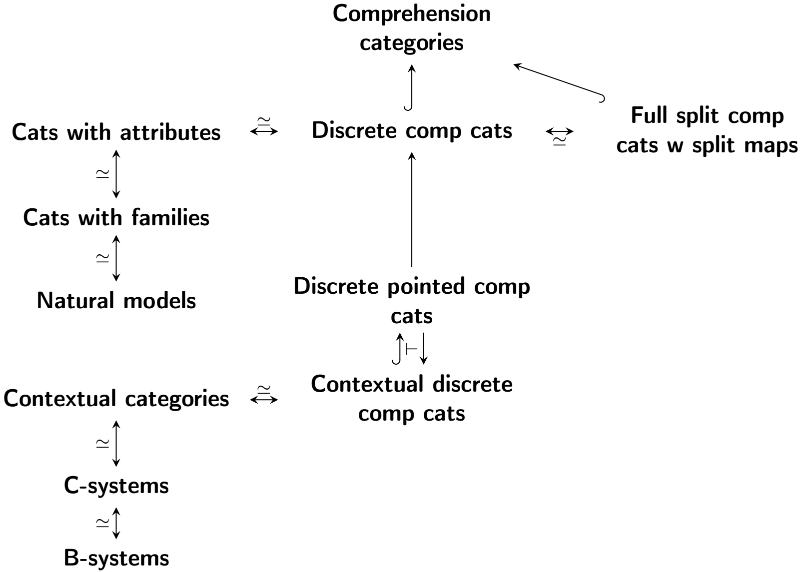
Construction

Given a clan $(\mathcal{C}, \mathcal{D})$, call an object $X \in \mathcal{C}$ **separated** if $X \to X \times X$ is a display map.

The full subcategory of separated objects is a finite-limit category.

Outline

Frameworks with types as primitive



Categories with families

Definition

A **category with families** consists of

1. a category $\mathcal{C}$;
2. a presheaf $\mathrm{Ty} : \mathcal{C}^{\mathsf{op}} \longrightarrow \mathsf{Set}$;
3. a presheaf $\mathrm{Tm} : \int_{\mathcal{C}} \mathrm{Ty} \longrightarrow \mathsf{Set}$; and
4. for each $\Gamma \in \mathcal{C}, A \in \mathrm{Ty}(\Gamma)$, an object $\Gamma.A$ and map $p_A : \Gamma.A \longrightarrow \Gamma$ representing $\mathrm{Tm}(\Gamma, A)$.

Why (not) use categories with families?

+ relatively close to syntax

Categories with families

Definition

- A **strict map**[a] is a functor and natural transformations preserving $\Gamma.A, p_A$ on the nose.
  - The 1-category whose 1-cells are strict maps is equivalent to the 1-category ofdiscrete comprehension categories.
- A **weak map**[b] preserves $\Gamma.A, p_A$ only up to isomorphism.
  - The 2-category whose 1-cells are weak maps is equivalent to the 2-category of full, split comprehension categories with split maps.
- A **pseudo map**[c] preserves reindexing only up to isomorphism.
  - The 2-category whose 1-cells are pseudo maps is equivalent to the 2-category of full, split discrete comprehension categories.

---

[a] Dybjer
[b] Birkedal, Clouston, Mannaa, Møgelberg, Pitts, Spitters
[c] Clairambault, Dybjer

Contextual categories

Definition

A **contextual category** consists of

1. a category $\mathcal{C}$ equipped with a distinguished terminal object 1;
2. a tree structure on $\mathrm{ob}\,\mathcal{C}$ with root 1;
3. for each non-root object A, a "projection" $p_A : A \longrightarrow \mathrm{par}(A)$ from A to its parent;
4. pullbacks of projections along arbitrary maps to projections $f^* p_A = p_{f^* A}$, strictly functorial in that $1^* A = A$, $(fg)^* A = g^* f^* A$.

Why (not) use contextual categories?

+ very close to syntax
− difficult to construct from mathematical objects

Contextual categories

Definition

Given a comprehension category $(\mathcal{C}, \mathcal{T}, p, \chi)$, the **contextual slice** $\mathcal{C} \mathbin{\downarrow} \Gamma$ is the comprehension category whose underlying category

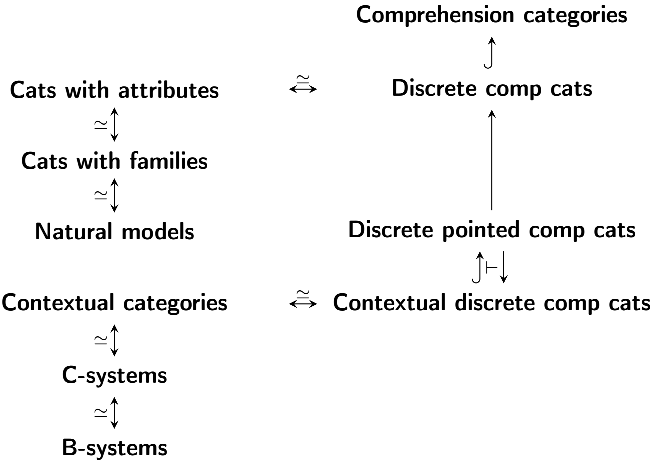
- has objects finite sequences $(A_0, \ldots, A_{n-1})$ in which $A_k \in \mathcal{T}_{\Gamma.A_0 \ldots A_{k-1}}$, for each $0 \leq k < n$
- has morphisms inherited from $\mathcal{C}$.

The comprehension category $(\mathcal{C}, \mathcal{T}, p, \chi)$ is **contextual** if $\mathcal{C}$ has a terminal object 1 and the projection $\mathcal{C} \mathbin{\downarrow} 1 \longrightarrow \mathcal{C}$ is bijective on objects.

Theorem

The 1-category of contextual comprehension categories and strict maps is equivalent to the 2-category of contextual comprehension categories and pseudo maps.

Frameworks with types as primitive



Outline

Summary

- Comprehension categories encompass other notions, which can be characterized via structure on comprehension categories
- Comparison on the level of 2-categories, considering pseudo-maps instead of strict maps
- Most notions are well-behaved in that they are equivalent to 2-categories of certain comprehension categories

Future work

- A textbook?
- An axiomatization?
  - Universal properties giving the equivalences?
  - Strict vs. weak models of 2-theories?

Thank you!