A Syntax for Strictly Associative and Unital $\infty$-Categories

Eric Finster  Alex Rice  Jamie Vicary

LICS 2024, Tallinn
CATT is a type theory for $\infty$-categories.
\texttt{CATT} is a type theory for $\infty$-categories.

Globular $\infty$-categories contain higher dimensional arrows.
\textbf{The Type Theory }\text{CATT}\textbf{ }

\text{CATT} is a type theory for $\infty$-categories.

Globular $\infty$-categories contain higher dimensional arrows.

Terms of \text{CATT} correspond to operations.
\textbf{CATT} is a type theory for \(\infty\)-categories.

Globular \(\infty\)-categories contain higher dimensional arrows.

Terms of \textbf{CATT} correspond to operations.

A type gives the boundary of a term.
Terms in CATT

Terms represent the possible operations in a globular $\infty$-category.

Terms built over \textit{pasting diagrams}.

Compound terms using substitutions.
Terms in **CAT**

Terms represent the possible operations in a globular $\infty$-category.

Terms built over *pasting diagrams*. Compound terms using substitutions.

\[
\text{coh} \left( \Gamma : s \to t \right)
\]
Terms in \texttt{CATT}

Terms represent the possible operations in a globular $\infty$-category.

Terms built over \textit{pasting diagrams}.

\[
\text{coh} (\Gamma : s \to t)
\]

\[
\Gamma := \begin{array}{c}
x \xrightarrow{f} y \xrightarrow{g} z \\
\end{array}
\]

\[
f \ast g := \text{coh} (\Gamma : x \to z)
\]

Compound terms using substitutions.

\[
\sigma : \{\begin{array}{c}
J \quad \sigma \\
K \\
\end{array}
\}
\]

\[
a \ast (b \ast c) := (f \ast g)
\]
Terms in \( \text{CAT} \)

Terms represent the possible operations in a globular \( \infty \)-category.

Terms built over *pasting diagrams.*

\[
\text{coh} \left( \Gamma : s \to t \right)
\]

\[
\Gamma := x \xrightarrow{f} y \xrightarrow{g} z
\]

\[
f \circ g := \text{coh} \left( \Gamma : x \to z \right)
\]

\[
\Delta := x \xleftarrow{f} y \xrightarrow{i} z
\]

\[
\alpha \otimes \beta := \text{coh} \left( \Delta : f \circ g \to h \circ i \right)
\]

Compound terms using substitutions.
Terms in $\mathbf{Catt}$

Terms represent the possible operations in a globular $\infty$-category.

Terms built over pasting diagrams.

$$\text{coh } (\Gamma : s \to t)$$

$$\Gamma := x \xrightarrow{f} y \xrightarrow{g} z$$

$$f \ast g := \text{coh } (\Gamma : x \to z)$$

Compound terms using substitutions.

$$s[\sigma]$$

$$\Delta := x \xleftarrow{f} y \xrightarrow{g} z$$

$$\alpha \otimes \beta := \text{coh } (\Delta : f \ast g \to h \ast i)$$
Terms in \textbf{CATT}

Terms represent the possible operations in a globular $\infty$-category.

Terms built over \textit{pasting diagrams}.

$$\text{coh} \ (\Gamma : s \to t)$$

$$\Gamma := x \xrightarrow{f} y \xrightarrow{g} z$$

$$f \ast g := \text{coh} \ (\Gamma : x \to z)$$

$$\Delta := x \xrightarrow{f} y \xrightarrow{g} z$$

$$\alpha \otimes \beta := \text{coh} \ (\Delta : f \ast g \to h \ast i)$$

Compound terms using substitutions.

$$s[\sigma]$$

$$\sigma := \langle f \mapsto a, g \mapsto (b \ast c) \rangle$$

$$a \ast (b \ast c) := (f \ast g)[\sigma]$$
\[ \Delta := x \xrightarrow{a} y \xrightarrow{b} z \xrightarrow{c} w \]
\[ \Delta := x \xrightarrow{a} y \xrightarrow{b} z \xrightarrow{c} w \]

\[ \alpha_{a,b,c} := \text{coh} (\Delta : (a \ast b) \ast c \to a \ast (b \ast c)) \]
The theory described by CATT is Weak.

Laws of categories are given by equivalence.
The theory described by CATT is Weak.

Laws of categories are given by equivalence.

Weak \leftrightarrow \text{Harder to use} \quad \text{Strict} \quad \text{Easier to use}
The theory described by $\text{CATT}$ is \textit{Weak}.

Laws of categories are given by equivalence.

\begin{center}
\begin{tikzpicture}[->,>=stealth',shorten >=1pt,semithick]
    \node (weak) at (0,0) {Weak};
    \node (strict) at (4,0) {Strict};
    \draw (weak) -- (strict);
\end{tikzpicture}
\end{center}

- Harder to use \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad Easier to use
- More expressive \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad Less expressive
The theory described by CATT is Weak.

Laws of categories are given by equivalence.

Weak ← Semistrict → Strict

Harder to use

More expressive

Easier to use

Less expressive
The theory described by $\text{CATT}$ is \textit{Weak}.

Laws of categories are given by equivalence.

\begin{center}
\begin{tikzpicture}
  \node (A) at (0,0) {Weak};
  \node (B) at (3.5,0) {Semistrict};
  \node (C) at (7,0) {Strict};
  \draw[->] (A) -- (B);
  \draw[->] (B) -- (C);
  \node (D) at (2.5,-2) {\texttt{CATT}_{\text{su}}};
  \draw[->] (D) -- (B);
  \node at (1.75,-4) {Harder to use};
  \node at (5.25,-4) {Easier to use};
  \node at (0.75,-5) {More expressive};
  \node at (5.75,-5) {Less expressive};
\end{tikzpicture}
\end{center}
The theory described by $C_{\text{ATT}}$ is Weak.

Laws of categories are given by equivalence.

\[
\begin{align*}
\text{Weak} & \leftarrow \text{Semistrict} \rightarrow \text{Strict} \\
& \downarrow \quad \downarrow \\
& C_{\text{ATT}_{\text{su}}} \quad C_{\text{ATT}_{\text{sua}}} \\
\text{Harder to use} & \quad \text{Easier to use} \\
\text{More expressive} & \quad \text{Less expressive}
\end{align*}
\]
(Semi)strictness allows more operations to be defined.
(Semi)strictness allows more operations to be defined.
(Semi)strictness allows more operations to be defined.

\[(\text{id}_a \otimes B) \ast (A \otimes \text{id}_d) : (a \ast b) \ast h \to f \ast (c \ast d)\]
$\text{CATT}_\text{sua}$ has trivial equality.

$\text{CATT}_\text{su}$ has disc removal, endo-coherence removal, and pruning.

In $\text{CATT}_\text{sua}$, pruning is replaced by insertion.

$$\text{CATT}_\text{sua} := \text{CATT} + \text{insertion} + \text{disc removal} + \text{endo-coherence removal}$$
Insertion Rule

\[(a * b) * c \equiv \text{coh} (x \overset{a}{\rightarrow} y \overset{b}{\rightarrow} z \overset{c}{\rightarrow} w : x \rightarrow w)\]
\[(a \ast b) \ast c =_{\text{sua}} a \ast b \ast c \equiv \text{coh} (x \rightarrow^a y \rightarrow^b z \rightarrow^c w : x \rightarrow w)\]

Recalling \((a \ast b) \ast c \equiv (f \ast g)[[f \mapsto a, g \mapsto b \ast c]]\):

\[
x \xrightarrow{f} y \xrightarrow{b \ast c} z
\]

\[
x' \xrightarrow{f'} y' \xrightarrow{g'} z'
\]

is sent to:

\[
x \xrightarrow{f} x' \xrightarrow{f'} y' \xrightarrow{g'} z'
\]
Universal Property of Insertion

\[
\text{coh } (\Delta : s \to t)[\sigma] \quad x \in \Delta \quad x[\sigma] \equiv \text{coh } (\Theta : u \to v)[\tau]
\]
Universal Property of Insertion

\[
\text{coh} \left( \Delta : s \to t \right)[\sigma] \quad x \in \Delta \quad x[\sigma] \equiv \text{coh} \left( \Theta : u \to v \right)[\tau] \quad (+ \text{syntactic side condition})
\]
Universal Property of Insertion

\[ \text{coh} (\Delta : s \to t)[\sigma] \quad x \in \Delta \quad x[\sigma] \equiv \text{coh} (\Theta : u \to v)[\tau] \quad (+ \text{syntactic side condition}) \]
Universal Property of Insertion

\[ \text{coh} (\Delta : s \to t)[\sigma] \quad x \in \Delta \quad x[\sigma] \equiv \text{coh} (\Theta : u \to v)[\tau] \quad (\text{+ syntactic side condition}) \]
Universal Property of Insertion

\[ \text{coh } (\Delta : s \to t)[\sigma] \quad x \in \Delta \quad x[\sigma] \equiv \text{coh } (\Theta : u \to v)[\tau] \quad (+ \text{ syntactic side condition}) \]
Universal Property of Insertion

\[
\text{coh } (\Delta : s \to t)[\sigma] \quad x \in \Delta \quad x[\sigma] \equiv \text{coh } (\Theta : u \to v)[\tau] \quad (+ \text{ syntactic side condition})
\]
Universal Property of Insertion

\[
\text{coh (} \Delta : s \rightarrow t)[\sigma] \quad x \in \Delta \quad x[\sigma] \equiv \text{coh (} \Theta : u \rightarrow v)[\tau] \quad (+ \text{syntactic side condition})
\]
Universal Property of Insertion

\[
\text{coh} (\Delta : s \to t)[\sigma] \quad x \in \Delta \quad x[\sigma] \equiv \text{coh} (\Theta : u \to v)[\tau] \quad (+ \text{syntactic side condition})
\]
Universal Property of Insertion

\[
\text{coh} (\Delta : s \to t)[\sigma] \quad x \in \Delta \quad x[\sigma] \equiv \text{coh} (\Theta : u \to v)[\tau] \quad (+ \text{sytactic side condition})
\]

\[
\begin{align*}
D & \xrightarrow{x} \Delta \\
\text{coh} (\Theta : u \to v) & \downarrow \\
\Theta & \xleftarrow{l} \Delta \ll_x \Theta \\
\Gamma & \\
\Delta \ll_x \Theta & \\
\end{align*}
\]

\[
\text{coh} (\Delta : s \to t)[\sigma] = \text{coh} (\Delta \ll_x \Theta : s[\kappa] \to t[\kappa])[\sigma \ll_x \tau]
\]
Normalisation

Equality in $\text{CATT}_{\text{sua}}$ is decidable.

$\text{CATT}_{\text{sua}}$ has unique normal forms.

Obtained by reduction system.

Termination: Assign syntactic complexity to terms.

Confluence: Encode various constructions in Agda.
Normalisation

Equality in $\text{CATT}_{\text{sua}}$ is decidable.

$\text{CATT}_{\text{sua}}$ has unique normal forms.

Obtained by reduction system.

Termination: Assign *syntactic complexity* to terms.

Confluence: Encode various constructions in Agda.

Type checking is decidable.

We provide a interpreter which:

- Provides tools for construction terms.
- Type checks terms.
- Reduces terms to $\text{CATT}_{\text{sua}}$ normal form.
We introduce the type theory $\text{CATT}_{\text{sua}}$.

$\text{CATT}_{\text{sua}}$ models strictly unital and associative $\infty$-categories.

$\text{CATT}_{\text{sua}}$ terms are simpler than their $\text{CATT}$ or $\text{CATT}_{\text{su}}$ equivalents.

Normal forms for $\text{CATT}_{\text{sua}}$ are obtained via a reduction system:

This reduction system is strongly terminating and confluent.

Try our interpreter for $\text{CATT}_{\text{sua}}$:

https://github.com/alexarice/catt-strict

Thank you for listening.