Generalized Set Theories in Isabelle/HOL

https://github.com/ultra-group/isabelle-gst

Ciarán M. Dunne www.macs.hw.ac.uk/~cmd1/
Joint work with: J. B. Wells
Heriot-Watt University, Scotland
2022-07-12

Types & Sets

- Isabelle/HOL: math-objects organized into lots of different types. e.g.,

  π :: real, $\quad \langle 5, 5 \rangle$ :: nat $*$ nat, $\quad (\lambda\, x\, .\, x = \pi)$ :: real $\Rightarrow$ bool

- What if we don't want this?
- Isabelle/ZF: implement math-objects as 'pure' sets living in one big type d. e.g.,
  - membership relation:

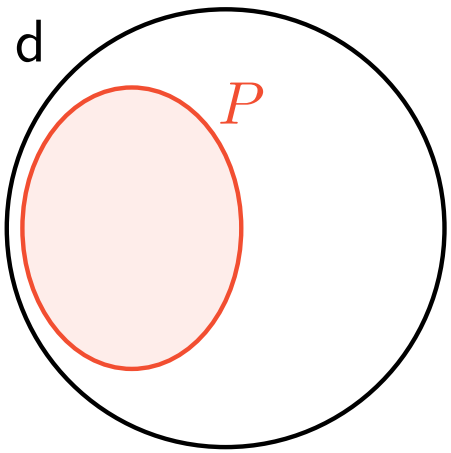
    $\in$ :: d $\Rightarrow$ d $\Rightarrow$ bool satisfying ZF axioms
  - natural numbers:

    $0 := \emptyset, \quad 1 := \{0\} = \{\emptyset\}, \quad 2 := \{0, 1\} = \{\emptyset, \{\emptyset\}\},$
    $3 := \{0, 1, 2\} = \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\},$
  - ordered pairs:

    $$\begin{aligned} \langle 0, 1 \rangle &:= \{\{0\}, \{0, 1\}\} \\ &= \{\{\emptyset\}, \{\emptyset, \{\emptyset\}\}\} \end{aligned}$$

Soft Types

- Most objects have the same type. e.g., $3 :: \mathsf{d}$, $\langle 0, 1 \rangle :: \mathsf{d}$.
- Keep track of meaning using soft types. (type-like predicates $P :: \mathsf{d} \Rightarrow \mathsf{bool}$, where $x : P$ abbreviates $P\,x$).



- e.g. $\emptyset : \mathsf{Set}$, $\langle 0, 1 \rangle : \mathsf{Nat} * \mathsf{Nat}$, $\mathcal{P} : \mathsf{SetOf\,Set} \rightarrowtriangle \mathsf{Set}$.
- But in ZF we have representation overlaps!
  e.g., both $3 : \mathsf{Nat}$ and $3 : \mathsf{Set}$ hold.

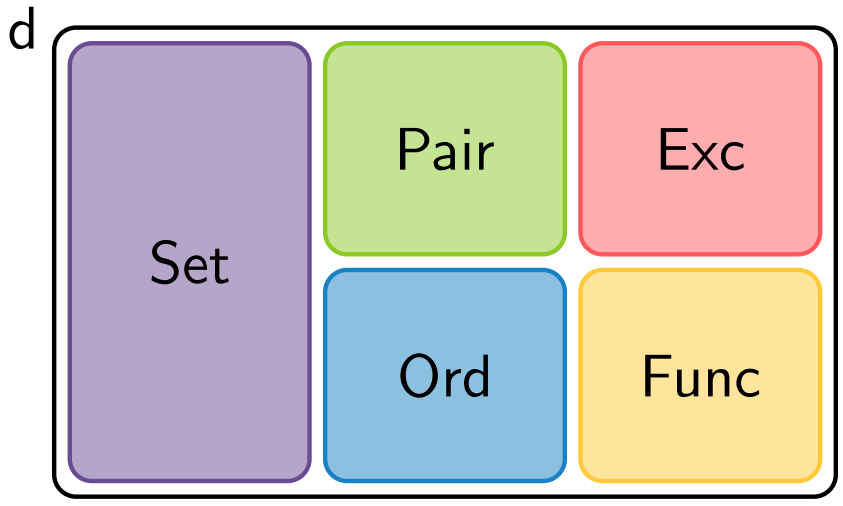
Searching for a Middle Path

Generalized Set Theories (GSTs) — HOL axioms s.t.

- avoid type system awkwardness:
  most math-objects live in a single domain type d.
- avoid representation overlap:
  sets, functions, numbers, pairs, etc. can all be distinct.
- allow support for partial functions/undefinedness:
  a special 'exception object' $\bullet :: \mathsf{d}$ inside the domain, where $\pi \div 0 = \bullet$, and $\{\bullet\} = \bullet$, etc.

GST Axioms

- Mix and match axioms features for different math-objects (sets, pairs, ordinals, functions, etc.)
- Extra axioms to ensure a partition on the type:
  - cover: $\mathsf{Set} \sqcup \mathsf{Pair} \sqcup \mathsf{Ord} \sqcup \mathsf{Func} \sqcup \mathsf{Exc} = \top$
  - disjoint: $\mathsf{Set} \sqcap \mathsf{Pair} = \bot$, $\mathsf{Set} \sqcap \mathsf{Ord} = \bot$, $\ldots$
    $\mathsf{Pair} \sqcap \mathsf{Ord} = \bot$, $\mathsf{Pair} \sqcap \mathsf{Func} = \bot$, $\ldots$, etc.



Example: Pointwise Subtraction

- We can define an overloaded subtraction operator on natural numbers $n :: \mathsf{d}$ and functions $f :: \mathsf{d}$, where $f : \mathsf{Nat} \to \mathsf{Nat}$.

$$(-) :: \mathsf{d} \Rightarrow \mathsf{d} \Rightarrow \mathsf{d}$$

- If $n, m : \mathsf{Nat}$,
  - $n \geq m$ implies $n - m : \mathsf{Nat}$,
  - $n < m$ implies $n - m = \bullet$
- If $f, g : \mathsf{Nat} \to \mathsf{Nat}$,
  - then $\forall\, b : \mathsf{Nat}\,.\,(f - g)(b) = f(b) - g(b)$

- Let $(x : \mathsf{Maybe}\, P) := ((x \neq \bullet \to x : P) \vee x = \bullet)$.
- We can prove the soft intersection type:

$$(-) : (\mathsf{Nat} \twoheadrightarrow \mathsf{Nat} \twoheadrightarrow \mathsf{Maybe}\,\mathsf{Nat}) \sqcap ((\mathsf{Nat} \to \mathsf{Nat}) \twoheadrightarrow (\mathsf{Nat} \to \mathsf{Nat}) \twoheadrightarrow (\mathsf{Nat} \to \mathsf{Maybe}\,\mathsf{Nat}))$$