Adding an Abstraction Barrier to ZF Set Theory

Conference for Intelligent Computer Mathematics 2020

Ciarán M. Dunne, J. B. Wells, and Fairouz Kamareddine

http://www.macs.hw.ac.uk/~cmd1/

Pre-print: https://arxiv.org/abs/2005.13954

July 27, 2020

Foundations of Mathematics

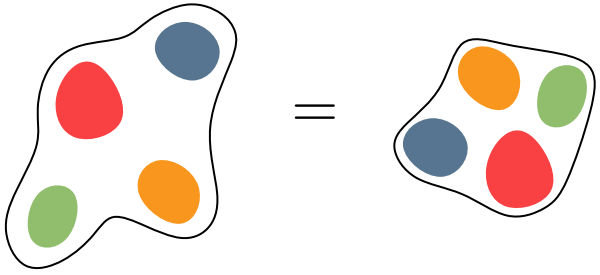
- Mathematical foundations are systems that use a small number of fundamental concepts to model the practice of mathematicians.
- Such systems provide rigorous accounts of mathematical objects such as numbers, vectors, functions, groups and their properties.
- Axiomatic set theories — particularly Zermelo-Frankel (ZF) with first-order logic (FOL) — are broadly accepted by mathematicians.
- Type theories are foundations of mathematics often used in machine-assisted theorem proving, and also in programming languages.
- Translation of mathematical text into a foundation is called formalisation.

Formalisation of Mathematics

- Foundations are idealisations of mathematical practice, so formalisation requires significant translation.
- Awkward compromises must constantly be made to fit the mathematics to the foundation.
- Current foundations fail to accurately capture the practice of mathematicians.
- We consider the consequences of founding a piece of mathematical text in ZF set theory to demonstrate one of these issues, and propose a solution to this particular issue.

Zermelo-Fraenkel Set Theory

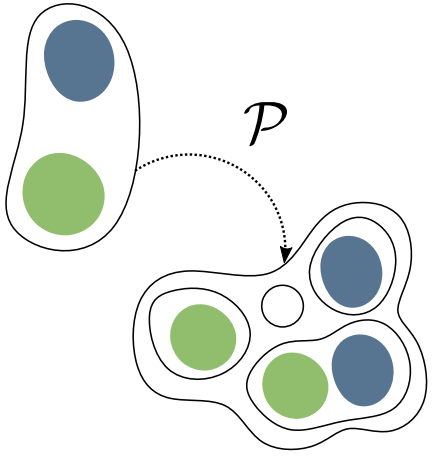
- ZF is a collection of axioms stated in first-order logic.
- Initial system proposed by Ernst Zermelo (above) in 1909, amended by Abraham Fraenkel (below), von Neumann and Skolem in 1922.
- The axioms describe a domain of objects called sets, and the behaviour of the set membership relation $\in$.
- Logical deduction from these axioms provides us with a rich theory of sets.
- Sets are used to represent mathematical objects, so that for many theorems about such objects there is a corresponding theorem about sets provable from ZF.



Axiom of Extensionality

$\forall x, y : (\forall a : a \in x \leftrightarrow a \in y) \rightarrow x = y$

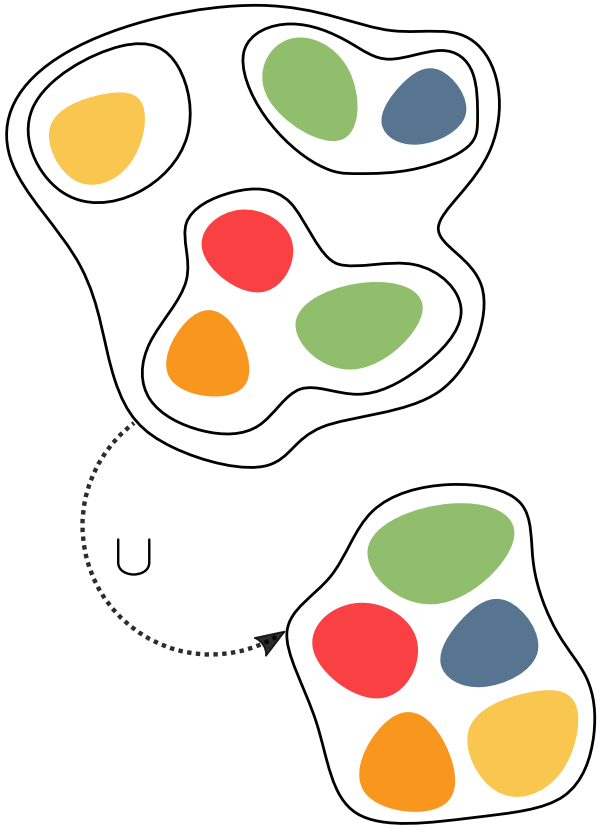
"If two sets contain exactly the same members, then they are equal."



Axiom of Power Set

$\forall x : \exists y : \forall z : z \in y \leftrightarrow z \subseteq x$

"The power set of a set exists."



Axiom of Union

$\forall x : \exists y : \forall a : a \in y \leftrightarrow (\exists z \in x : a \in z)$

"The union of all sets inside a set exists."

Axiom Schema of Replacement

$$\forall\, c_1, c_2, x : (\forall\, a \in x : \exists!\, b : \hat{\varphi}(a, b)) \to (\exists\, y : \forall\, b : b \in y \leftrightarrow \exists\, a \in x : \hat{\varphi}(a, b))$$

"For any formula $\hat{\varphi}(a, b)$ with at most c_1, c_2, a, b free, if for each $a \in x$ there is a unique b such that $\hat{\varphi}(a, b)$ holds, then the set containing all such b exists''

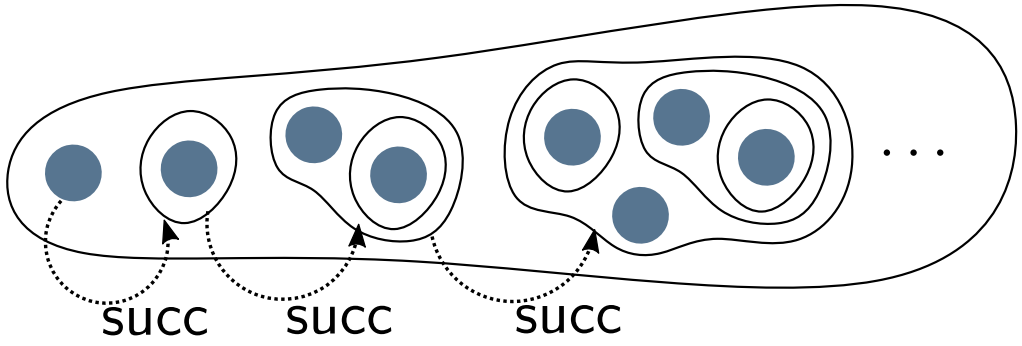
- Used to define set-builder notation:
  $\{\, y \mid \exists\, x \in B : \hat{\varphi}(x, y) \,\}$
- Allows us to prove the existence of $\emptyset$, $\{X, Y\}$, $\{\, a \in X \mid \psi \,\}$.
- Needed for results in set theory, as well as results such as the Borel determinacy theorem.

Axiom of Foundation

$$\forall\, x : x = \emptyset \vee (\exists\, y \in x : y \cap x = \emptyset)$$

"Every set is well-founded."

- Rules out infinitely descending chains of sets $X_0 \ni X_1 \ni \cdots$.
  - e.g. no self-containing sets $X \in X$.



Axiom of Infinity

$$\exists y : \emptyset \in y \wedge (\forall x \in y : \text{succ}(x) \in y)$$

$$\text{succ}(x) := x \cup \{x\}$$

"There exists a set containing $\emptyset$, closed under the successor operation."

- The von Neumann natural numbers can be defined as:

$$0 := \emptyset,\ 1 := \{0\},\ 2 := \{0, 1\},\ 3 := \{0, 1, 2\},\ \ldots$$

- Infinity proves the existence of $\mathbb{N}$, the natural numbers.
- The set membership relation $\in$ acts like the $<$ ordering on $\mathbb{N}$.

Ordered Pairs in Set Theory

- An ordered pair (a, b) is a container that holds two objects.
- Any definition of ordered pairs must satisfy:

Characteristic Property of Ordered Pairs

$$(a, b) = (c, d) \text{ if and only if } a = c \text{ and } b = d.$$

- Projection relations or functions – π_1, π_2 – are used to reason about the contents.

  For example: $3\ \pi_1\ (3, 2)$

Definition

Widely used set-theoretic definition given by Kuratowski in 1921:

$$\langle a, b \rangle = \{\{a\}, \{a, b\}\}$$

Functions in ZF

- Relations and functions are defined as sets of ordered pairs.
- For relations, we define $x\,R\,y := (x, y) \in R$.
- For functions, $f(x)$ is the unique y such that $(x, y) \in f$.
  So for each x there must be at most one value y for which $(x, y) \in f$.

Example

A function that satisfies the formula $\forall\, x \in \mathbb{N} : f(x) = x^2$.

$f := \{\, \langle x, x^2 \rangle \mid x \in \mathbb{N} \,\} = \{\langle 0, 0\rangle, \langle 1, 1\rangle, \langle 2, 4\rangle, \langle 3, 9\rangle, \ldots\}$

Functions in ZF

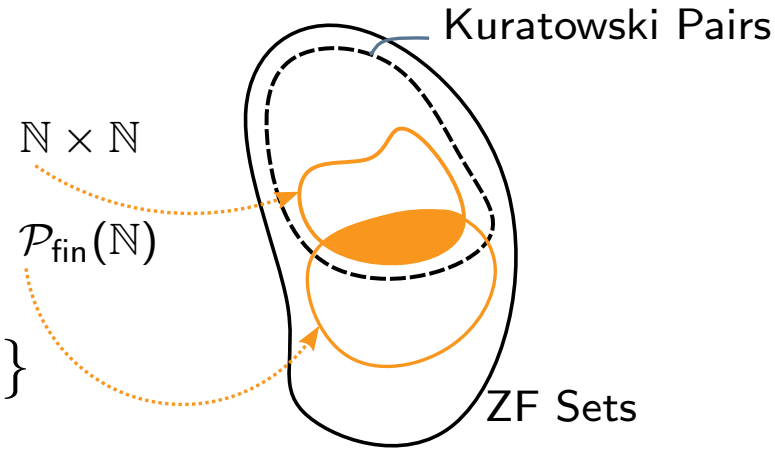
- A user may want to define a function acting on a domain containing sets and ordered pairs which satisfies the following logical constraints:
  - $D = (\mathbb{N} \times \mathbb{N}) \cup \mathcal{P}_{\text{fin}}(\mathbb{N})$
  - $g : D \to \mathbb{N}$
  - $g(x) = \begin{cases} p + q & \text{if } x = \langle p, q \rangle \text{ for some } p, q \in \mathbb{N} \\ \sum_{i=1}^{n} p_i & \text{if } x = \{p_1, \ldots, p_n\} \subseteq \mathbb{N} \end{cases}$
- Does ZF provide a function g satisfying the specification?
- What is $g(\langle 0, 1 \rangle)$? $g(\{1, 2\})$?

Functions in ZF

- From the specification, we might expect that $g(\langle 0, 1\rangle) = 1$, and $g(\{1, 2\}) = 3$.
- However, by definition of Kuratowski ordered pairs: $\langle 0, 1\rangle = \{\{0\}, \{0, 1\}\}$.
- By definition of von Neumann natural numbers: $\{1, 2\} = \{\{0\}, \{0, 1\}\}$.
- So by definition of function, there must be only one y such that $\langle \{\{0\}, \{0, 1\}\}, y\rangle \in g$, and hence only one result for $g(\langle 0, 1\rangle)$ and $g(\{1, 2\})$.
- What went wrong?

- The behaviour of the specified function is based on cases of the 'type' of the object, which is only visible at the meta-level.
- We assumed the cases were mutually exclusive, because $\mathbb{N} \times \mathbb{N}$ and $\mathcal{P}_{\text{fin}}(\mathbb{N})$ are considered to be disjoint, as ordered pairs are not usually thought of as sets.
- We are using a naive translation of mathematical text to ZF sets, roughly:

$$\begin{aligned} \{b_1, \ldots, b_n\}^* &= \{b_1^*, \ldots, b_n^*\} \\ (b, c)^* &= \{\{b^*\}, \{b^*, c^*\}\} \\ (n+1)^* &= n^* \cup \{n^*\} \end{aligned}$$



- Reasoning on a mixture of sets and objects represented by ZF sets requires us to ensure the collections of ZF sets in use are disjoint for each 'type'.

Types?

- In type-theoretic foundations, every typeable term x has at least one type τ, written $x : \tau$.

Example

$$42 : \mathsf{Nat} \qquad (2, 5) : \mathsf{Nat} * \mathsf{Nat} \qquad (\cdot)^2 : \mathsf{Nat} \to \mathsf{Nat}$$

- Types are used to organise the operations that may be performed on objects, and characterise the objects that can exist.
- Overloading of operations can be safely performed since type information can often be inferred by an algorithm.

Maybe Not...

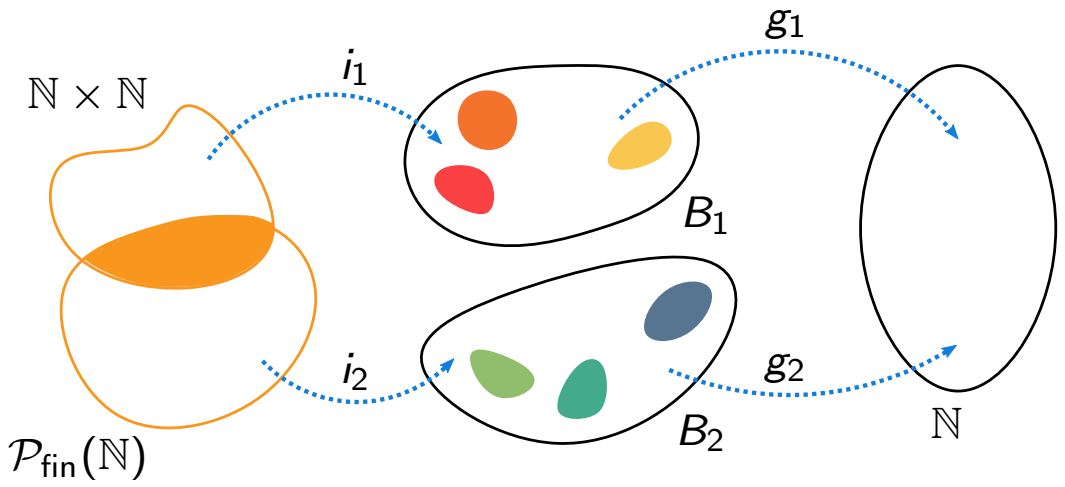
- Overloading allows us to define $g_1 : (\mathsf{Nat} * \mathsf{Nat}) \to \mathsf{Nat}$ and $g_2 : \mathcal{P}(\mathsf{Nat}) \to \mathsf{Nat}$, and then:

$$g(x) = \begin{cases} g_1(x) & \text{if the type of } x \text{ is } \mathsf{Nat} * \mathsf{Nat} \\ g_2(x) & \text{if the type of } x \text{ is } \mathcal{P}(\mathsf{Nat}) \end{cases}$$

- However, the typing rules are often very complicated, and require machine assistance to carry them out.
- A large portion of mathematics is written as if it is founded in set theory. Formulating it in type theory may be confusing for mathematicians.

Tagging in Set Theory

- What if we want to stay within set theory?



Definition

$$i_1 : \mathbb{N} \times \mathbb{N} \to B_1 \qquad i_2 : \mathcal{P}_{\text{fin}}(\mathbb{N}) \to B_2$$

such that i_1, i_2 are injective and $B_1 \cap B_2 = \emptyset$.

Definition

$$g_1 : B_1 \to \mathbb{N} \qquad g_2 : B_2 \to \mathbb{N}$$

$$g_1(i_1(\langle p, q \rangle)) = p + q \qquad g_2(i_2(\{p_1, \ldots, p_n\})) = \sum_{i=1}^{n} p_i$$

Tagging: Results

- The function g can then be defined as $g_1 \cup g_2$ with domain $B_1 \cup B_2$, and the user may apply i_1 and i_2 tell g whether to use g_1 or g_2.
- The mappings $x \mapsto \langle 0, x \rangle$, $x \mapsto \langle 1, x \rangle$ are often used for i_1, i_2.
- Tagging is ugly and annoying, as we would like to reason about sets and pairs without the use of tags.
- If you don't want to do tagging, there isn't much choice for resolving this issue whilst staying within set-theoretic foundations.

What else?

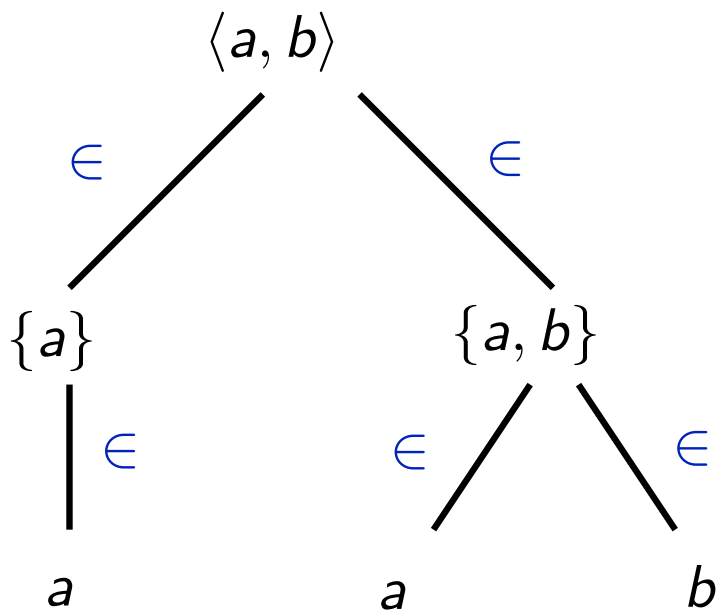
- Some set theories have non-set objects called atoms or urelements, though they usually have no relational structure.
- Long term goal: a set theory with an information hiding mechanism in which the user can specify a set-theoretic representation of a class of mathematical objects, and hide the gritty details using genuine non-sets.
- These special non-sets will have the properties of a set representation provided by the user, but without the properties the user considers irrelevant.
- As a first instance of this aim, we arrive at ZFP (ZF with Ordered Pairs), which hides the representation of ordered pairs.

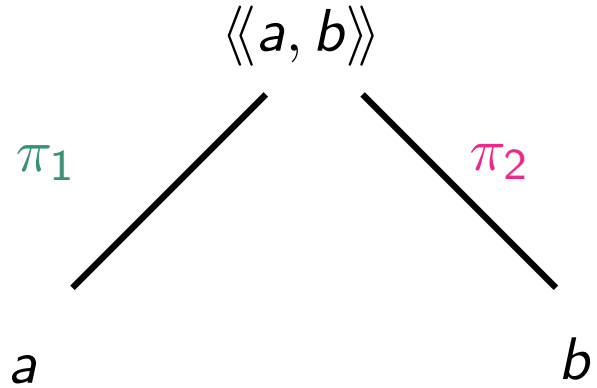
ZFP: Zermelo-Fraenkel Set Theory with (Ordered) Pairs

Introduction to ZFP

- ZFP extends ZF with ordered pairs as non-set objects.
- The domain of objects is split into pairs and sets.
- For any objects A, B, there exists a non-set ordered pair $\langle\langle A, B\rangle\rangle$, i.e. $\forall c : c \notin \langle\langle A, B\rangle\rangle$.
- Gained by modifying ZF's axioms to allow non-sets, and axiomatising relation symbols π_1, π_2 with desired properties.
- So that $\langle\langle A, B\rangle\rangle$ is the unique p such that $A\ \pi_1\ p$ and $B\ \pi_2\ p$.
- The domain of ZFP contains all of the pure sets of ZF, including Kuratowski pairs $\langle a, b\rangle$.

Diagram





Facts

- Sets can have pairs as their members, and pairs can have sets as their projections.
- Cartesian product can defined using Replacement nested inside Replacement:

$$A \times B = \cup \{ z \mid \exists c \in A : z = \{ p \mid \exists d \in B : p = \langle\langle c, d \rangle\rangle \} \}$$

- If $p = \langle\langle a, b \rangle\rangle$, then $\mathcal{P}(p) = \emptyset$ and $\cup p = \emptyset$, etc.

Consistency of ZFP

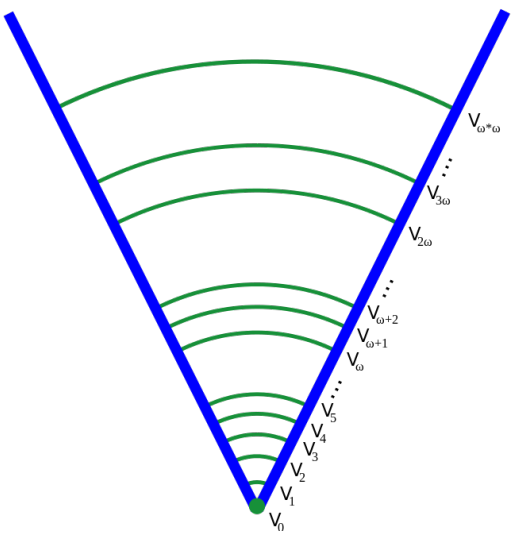
- The axioms of ZF are believed to be consistent.
- That is, people have tried to prove a formula φ and its negation $\neg\varphi$ in ZF for over a century to no avail.
- Can the same be said about ZFP?
- We can build a model of ZFP the sets of ZF.
- This provides us with a method of interpreting the formulas of ZFP within ZF.
- So if ZF is consistent, so is ZFP.

Models of ZF

- The Von-Neumann hierarchy and the membership relation $(V, \in)$ form a model of ZF.
- Generated via transfinite recursion by iterating the power set operator from $\emptyset$.

Von-Neumann Hierarchy:

$$V_0 = \emptyset,\ V_1 = \{\emptyset\},\ V_2 = \{\emptyset, \{\emptyset\}\},\ V_3 = \{\emptyset, \{\emptyset\}, \{\{\emptyset\}\}, \{\emptyset, \{\emptyset\}\}\},$$
$$\ldots,\ V_\lambda = \cup_{\beta<\lambda} V_\beta$$



- V_6 contains 2^{65536} elements, greater than the number of atoms in the known universe...
- Whilst everything is constructed out of the empty set, it still satisfies the axioms of ZF.

A Model of ZFP

- To build a model of ZFP in ZF, we iterate the power set and cartesian product operators.
- Objects need to be tagged at every stage to ensure a disjoint domain of pairs and sets.
- If $x = \langle 0, x' \rangle$, then x is a 'set'.
  If $x = \langle 1, \langle a, b \rangle \rangle$, then x is a 'pair'.
- The relations $\widehat{\in}, \widehat{\pi}_1, \widehat{\pi}_2$ act on $\mathbf{W}$ as they do in ZFP.

Definition

$$
\begin{aligned}
W_0 &= \emptyset \\
W_{\beta+1} &= (\{0\} \times \mathcal{P}(W_\beta)) \cup (\{1\} \times {W_\beta}^2) \\
W_\lambda &= \bigcup_{\beta \in \lambda} W_\beta
\end{aligned}
$$

Definition

$$
\begin{aligned}
x \mathrel{\widehat{\in}} y &:= \exists\, y' : y = \langle 0, y' \rangle \wedge x \in y' \\
x \mathrel{\widehat{\pi}_1} p &:= \exists\, y : p = \langle 1, \langle x, y \rangle \rangle \\
y \mathrel{\widehat{\pi}_2} p &:= \exists\, x : p = \langle 1, \langle x, y \rangle \rangle
\end{aligned}
$$

Interpretation of ZFP in ZF

- Any formula of ZFP φ can be interpreted in ZF as φ^* by restricting quantification to $\mathbf{W}$ and replacing occurrences of relation symbols with $\widehat{\in}, \widehat{\pi}_1, \widehat{\pi}_2$.
- For each axiom ϕ, the proof for ϕ^* has been machine-checked in Isabelle/ZF.[1]
- If ZFP $\vdash \varphi$, then ZF $\vdash \varphi^*$
- ZFP can be seen to be an axiomatisation of a fully tagged domain, with all of the nasty details abstracted away.

[1] See `http://www.macs.hw.ac.uk/~cmd1/cicm2020/ZFP.thy` for the source, and `http://www.macs.hw.ac.uk/~cmd1/cicm2020/ZFPDoc/index.html` for the HTML.

Future work: Information Hiding Mechanism

- What if we want more kinds of non-sets other than ordered pairs?
- ZFP depends on the consistency of ZF, and its ability to provide a representation for ordered pairs, and a tagging mechanism.
- The methods used for gaining primitive ordered pairs could be carried out for numbers, functions, and other mathematical objects.
- We propose designing a set theory which provides a mechanism for this, allowing the user to form new non-set objects with any desired properties, provided a valid representation exists.
- Relative consistency can be proved by showing that all of this can be simulated in ZF.

Conclusion

- ZF models mathematical text by representing mathematical objects as sets.
- Reasoning performed based on cases of the 'type' of object requires a messy translation to be considered founded in ZF.
- Types may not be the solution to this, due to complicated typing rules.
- Representation of ordered pairs can be hidden using ZFP.
- A generalisation of these methods may allow a theory of mathematical objects in the style of set theory in which representations are hidden.

- **Sets**:
  1. Set Extensionality: $\forall_{\mathsf{Set}}\, x, y : (\forall\, a : a \in x \leftrightarrow a \in y) \rightarrow x = y$
  2. Union: $\forall_{\mathsf{Set}}\, x : \exists\, y : \forall\, a : a \in y \leftrightarrow (\exists\, z \in x : a \in z)$
  3. Power Set: $\forall_{\mathsf{Set}}\, x : \exists\, y : \forall\, z : z \in y \leftrightarrow z \subseteq x$
  4. Infinity (ugly version): $\exists\, y : (\exists_{\mathsf{Set}}\, z \in y : \forall\, b : b \notin z) \wedge$
     $(\forall\, x \in y : \exists\, s \in y : \forall\, c : c \in s \leftrightarrow (c \in x \vee c = x))$.
  5. Replacement:
     $\forall\, c_1, c_2, x : (\forall\, a \in x : \exists!\, b : \varphi) \rightarrow (\exists_{\mathsf{Set}}\, y : \forall\, b : b \in y \leftrightarrow \exists\, a \in x : \varphi)$
  6. Foundation:
     $\forall_{\mathsf{Set}}\, x : x = \emptyset \vee (\exists\, a \in x : \neg \exists\, b \in x : b\ \pi_1\ a \vee b\ \pi_2\ a \vee b \in a)$
- **Ordered Pairs:**
  1. Ordered Pair Emptiness: $\forall_{\mathsf{Pair}}\, p : \forall\, a : a \notin p$
  2. Ordered Pair Formation: $\forall\, a, b : \exists\, p : a\ \pi_1\ p \wedge b\ \pi_2\ p$
  3. Projection Both-Or-Neither: $\forall\, p : (\exists\, a : a\ \pi_1\ p) \leftrightarrow (\exists\, b : b\ \pi_2\ p)$
  4. Projection Uniqueness: $\forall_{\mathsf{Pair}}\, p : (\exists!\, a : a\ \pi_1\ p) \wedge (\exists!\, b : b\ \pi_2\ p)$
  5. Ordered Pair Extensionality:
     $\forall_{\mathsf{Pair}}\, p, q : (\forall\, a : (a\ \pi_1\ p \leftrightarrow a\ \pi_1\ q) \wedge (a\ \pi_2\ p \leftrightarrow a\ \pi_2\ q)) \rightarrow p = q$