# Structuralism, Invariance, and Univalence

Steve Awodey

Institute for Advanced Study
and
Carnegie Mellon University

CSPM, Toulouse, April 2013

# Introduction

Advances in foundations have led to some developments that are significant for the philosophy of mathematics, particularly structuralism:

- Martin-Löf type theory can be interpreted into homotopy theory.

# Introduction

Advances in foundations have led to some developments that are significant for the philosophy of mathematics, particularly structuralism:

- Martin-Löf type theory can be interpreted into homotopy theory.
- Gives a new approach to the foundations of mathematics with intrinsic geometric content and a computational implementation.

# Introduction

Advances in foundations have led to some developments that are significant for the philosophy of mathematics, particularly structuralism:

- Martin-Löf type theory can be interpreted into homotopy theory.
- Gives a new approach to the foundations of mathematics with intrinsic geometric content and a computational implementation.
- Voevodsky has introduced a new axiom with both geometric and logical significance: the Univalence Axiom.

# Introduction

Advances in foundations have led to some developments that are significant for the philosophy of mathematics, particularly structuralism:

- Martin-Löf type theory can be interpreted into homotopy theory.
- Gives a new approach to the foundations of mathematics with intrinsic geometric content and a computational implementation.
- Voevodsky has introduced a new axiom with both geometric and logical significance: the Univalence Axiom.
- UA captures a familiar aspect of informal mathematical practice: one can identify isomorphic objects.

# Introduction

Advances in foundations have led to some developments that are significant for the philosophy of mathematics, particularly structuralism:

- Martin-Löf type theory can be interpreted into homotopy theory.
- Gives a new approach to the foundations of mathematics with intrinsic geometric content and a computational implementation.
- Voevodsky has introduced a new axiom with both geometric and logical significance: the Univalence Axiom.
- UA captures a familiar aspect of informal mathematical practice: one can identify isomorphic objects.
- It is incompatible with conventional foundations, but is a powerful addition to homotopy type theory.

# I. The Principle of Structuralism

The following statement may be called the *Principle of Structuralism*:

$$\text{Isomorphic objects are identical.} \qquad \text{(PS)}$$

# I. The Principle of Structuralism

The following statement may be called the *Principle of Structuralism*:

Isomorphic objects are identical. (PS)

From one perspective, this captures a principle of reasoning embodied in everyday mathematical practice:

# I. The Principle of Structuralism

The following statement may be called the *Principle of Structuralism*:

$$\text{Isomorphic objects are identical.} \tag{PS}$$

From one perspective, this captures a principle of reasoning embodied in everyday mathematical practice:

- the group $\mathbb{Z}$ is isomorphic to its own group of automorphisms $\mathrm{Aut}(\mathbb{Z})$, so these are *the same* group, namely the free group on one generator, $F(1)$.

# I. The Principle of Structuralism

The following statement may be called the *Principle of Structuralism*:

$$\text{Isomorphic objects are identical.} \qquad \text{(PS)}$$

From one perspective, this captures a principle of reasoning embodied in everyday mathematical practice:

- the group $\mathbb{Z}$ is isomorphic to its own group of automorphisms $\mathrm{Aut}(\mathbb{Z})$, so these are *the same* group, namely the free group on one generator, $F(1)$.
- The Cauchy reals are isomorphic to the Dedekind reals, so these are *the same* number field, $\mathbb{R}$.

# I. The Principle of Structuralism

The following statement may be called the *Principle of Structuralism*:

$$\text{Isomorphic objects are identical.} \qquad \text{(PS)}$$

From one perspective, this captures a principle of reasoning embodied in everyday mathematical practice:

- the group $\mathbb{Z}$ is isomorphic to its own group of automorphisms $\mathrm{Aut}(\mathbb{Z})$, so these are *the same* group, namely the free group on one generator, $F(1)$.
- The Cauchy reals are isomorphic to the Dedekind reals, so these are *the same* number field, $\mathbb{R}$.
- the unit interval $[0,1]$ is homeomorphic to the closed interval $[0,2]$, so these are *the same* topological space, say $I$.

# I. The Principle of Structuralism

Within a theorem or proof, it makes no difference which of two different "isomorphic copies" are used, and so they can be treated as *the same* mathematical object.

# I. The Principle of Structuralism

Within a theorem or proof, it makes no difference which of two different "isomorphic copies" are used, and so they can be treated as *the same* mathematical object.
This is sometimes referred to as "abuse of notation" or "working up to isomorphism".

# I. The Principle of Structuralism

Within a theorem or proof, it makes no difference which of two different "isomorphic copies" are used, and so they can be treated as *the same* mathematical object.
This is sometimes referred to as "abuse of notation" or "working up to isomorphism".
It is quite useful in practice, but it is literally false: it is incompatible with conventional foundations of mathematics in set theory.

# I. The Principle of Structuralism

Within a theorem or proof, it makes no difference which of two different “isomorphic copies” are used, and so they can be treated as *the same* mathematical object.
This is sometimes referred to as “abuse of notation” or ”working up to isomorphism”.
It is quite useful in practice, but it is literally false: it is incompatible with conventional foundations of mathematics in set theory.
The group of automorphisms of $\mathbb{Z}$ consists of functions, not numbers. The Cauchy reals are sequences of rationals, i.e. functions $\mathbb{N} \longrightarrow \mathbb{Q}$, and the Dedekind reals are subsets of rationals.

# I. The Principle of Structuralism

Within a theorem or proof, it makes no difference which of two different “isomorphic copies” are used, and so they can be treated as *the same* mathematical object.
This is sometimes referred to as “abuse of notation” or ”working up to isomorphism”.
It is quite useful in practice, but it is literally false: it is incompatible with conventional foundations of mathematics in set theory.
The group of automorphisms of $\mathbb{Z}$ consists of functions, not numbers. The Cauchy reals are sequences of rationals, i.e. functions $\mathbb{N} \longrightarrow \mathbb{Q}$, and the Dedekind reals are subsets of rationals.
Mathematical objects are often constructed out of other ones, and so also have some residual structure: i.e. the real number field actually consisting of functions or of subsets. So there is a clear sense in which (PS) is simply false.

# I. The Principle of Structuralism

What if we try to be generous and read "identity" in a weaker way:

"$A$ is identical to $B$" means "$A$ and $B$ have all the same properties".

# I. The Principle of Structuralism

What if we try to be generous and read "identity" in a weaker way:

"$A$ is identical to $B$" means "$A$ and $B$ have all the same properties".

We may want to restrict the properties that can occur on the right, in order for this to really be a weaker the condition.

So we have the reformulation:

$$A \cong B \Leftrightarrow \text{ for all relevant properties } P,\ P(A) \Rightarrow P(B) \qquad \text{(PS')}$$

# I. The Principle of Structuralism

However, for example, consider perfectly reasonable set-theoretic properties like:

$$\cap X \in X, \quad \text{or} \quad \emptyset \in X.$$

# I. The Principle of Structuralism

However, for example, consider perfectly reasonable set-theoretic properties like:

$$\cap X \in X, \quad \text{or} \quad \emptyset \in X.$$

These simple properties of sets do not “respect isomorphism” in the sense stated in (PS’).

# I. The Principle of Structuralism

However, for example, consider perfectly reasonable set-theoretic properties like:

$$\cap X \in X, \quad \text{or} \quad \emptyset \in X.$$

These simple properties of sets do not "respect isomorphism" in the sense stated in (PS').
So the task of determining the "relevant" properties is no simple matter.

# I. The Principle of Structuralism

However, for example, consider perfectly reasonable set-theoretic properties like:

$$\cap X \in X, \quad \text{or} \quad \emptyset \in X.$$

These simple properties of sets do not "respect isomorphism" in the sense stated in (PS').

So the task of determining the "relevant" properties is no simple matter.

Our leading question may be formulated as follows:

*Is there some sense in which (PS) can be held true?*

# I. The Principle of Structuralism

However, for example, consider perfectly reasonable set-theoretic properties like:

$$\cap X \in X, \quad \text{or} \quad \emptyset \in X.$$

These simple properties of sets do not "respect isomorphism" in the sense stated in (PS').
So the task of determining the "relevant" properties is no simple matter.
Our leading question may be formulated as follows:

*Is there some sense in which (PS) can be held true?*

This would legitimize mathematical practice and support structuralism.

# I. The Principle of Structuralism

However, for example, consider perfectly reasonable set-theoretic properties like:

$$\cap X \in X, \quad \text{or} \quad \emptyset \in X.$$

These simple properties of sets do not "respect isomorphism" in the sense stated in (PS').
So the task of determining the "relevant" properties is no simple matter.
Our leading question may be formulated as follows:

*Is there some sense in which (PS) can be held true?*

This would legitimize mathematical practice and support structuralism.
Let us consider the terms of (PS) in turn.

# II. Isomorphism

What does it mean to say that two things are isomorphic?

# II. Isomorphism

What does it mean to say that two things are isomorphic?
Some might be tempted to say:

$$A \cong B \;\Leftrightarrow\; A \text{ and } B \text{ have the same structure.} \tag{1}$$

# II. Isomorphism

What does it mean to say that two things are isomorphic?
Some might be tempted to say:

$$A \cong B \;\Leftrightarrow\; A \text{ and } B \text{ have the same structure.} \tag{1}$$

That is true, but it is not the *definition* of isomorphism.

# II. Isomorphism

What does it mean to say that two things are isomorphic?
Some might be tempted to say:

$$A \cong B \;\Leftrightarrow\; A \text{ and } B \text{ have the same structure.} \tag{1}$$

That is true, but it is not the *definition* of isomorphism.
It presumes the notion of structure and an identity criterion for it, and that is putting the cart before the horse.

# II. Isomorphism

What does it mean to say that two things are isomorphic?
Some might be tempted to say:

$$A \cong B \;\Leftrightarrow\; A \text{ and } B \text{ have the same structure.} \tag{1}$$

That is true, but it is not the *definition* of isomorphism.
It presumes the notion of structure and an identity criterion for it, and that is putting the cart before the horse.
*Structure* is like *color* or *the direction of a line* or *number*: it is an abstract concept.

# II. Isomorphism

What does it mean to say that two things are isomorphic?
Some might be tempted to say:

$$A \cong B \;\Leftrightarrow\; A \text{ and } B \text{ have the same structure.} \tag{1}$$

That is true, but it is not the *definition* of isomorphism.
It presumes the notion of structure and an identity criterion for it, and that is putting the cart before the horse.
*Structure* is like *color* or *the direction of a line* or *number*: it is an abstract concept.
As Frege taught us, it is determined by an abstraction principle, in this case:

$$\mathrm{str}(A) = \mathrm{str}(B) \;\Leftrightarrow\; A \cong B. \tag{DS}$$

# II. Isomorphism

What does it mean to say that two things are isomorphic?
Some might be tempted to say:

$$A \cong B \;\Leftrightarrow\; A \text{ and } B \text{ have the same structure.} \tag{1}$$

That is true, but it is not the *definition* of isomorphism.
It presumes the notion of structure and an identity criterion for it, and that is putting the cart before the horse.
*Structure* is like *color* or *the direction of a line* or *number*: it is an abstract concept.
As Frege taught us, it is determined by an abstraction principle, in this case:

$$\mathrm{str}(A) = \mathrm{str}(B) \;\Leftrightarrow\; A \cong B. \tag{DS}$$

The structure of $A$ is the same as the structure of $B$ just in case $A$ and $B$ are isomorphic. So (1) is true, but it's the definition of "structure" in terms of isomorphism, not the other way around.

# II. Isomorphism

So what is the definition of "isomorphism"?

# II. Isomorphism

So what is the definition of "isomorphism"?

$A$ and $B$ are *isomorphic*, $A \cong B$, if there are structure-preserving maps

$$f : A \longrightarrow B \quad \text{and} \quad g : B \longrightarrow A$$

such that

$$g \circ f = 1_A \quad \text{and} \quad f \circ g = 1_B.$$

# II. Isomorphism

So what is the definition of "isomorphism"?
$A$ and $B$ are *isomorphic*, $A \cong B$, if there are structure-preserving maps

$$f : A \longrightarrow B \quad \text{and} \quad g : B \longrightarrow A$$

such that

$$g \circ f = 1_A \quad \text{and} \quad f \circ g = 1_B.$$

This makes reference to "structure-preserving maps", "composition" $\circ$, and "identity" maps $1_A$, but these are primitive concepts of category theory, and need not be further defined.

# II. Isomorphism

So what is the definition of "isomorphism"?
$A$ and $B$ are *isomorphic*, $A \cong B$, if there are structure-preserving maps

$$f : A \longrightarrow B \quad \text{and} \quad g : B \longrightarrow A$$

such that

$$g \circ f = 1_A \quad \text{and} \quad f \circ g = 1_B.$$

This makes reference to "structure-preserving maps", "composition" $\circ$, and "identity" maps $1_A$, but these are primitive concepts of category theory, and need not be further defined.
Note that "isomorphism" is always relative to a given category, which determines a kind of structure.

# II. Isomorphism

Sometimes structures are built up from other ones, like groups, graphs, spaces. These are sets equipped with operations, relations, etc.

# II. Isomorphism

Sometimes structures are built up from other ones, like groups, graphs, spaces. These are sets equipped with operations, relations, etc.

The notion of a structure-preserving map, or *morphism*, is then defined in terms of maps in the underlying category of sets (i.e. functions) that preserve the operations, relations, etc. – i.e. "homomorphisms," in the usual sense.

# II. Isomorphism

Sometimes structures are built up from other ones, like groups, graphs, spaces. These are sets equipped with operations, relations, etc.

The notion of a structure-preserving map, or *morphism*, is then defined in terms of maps in the underlying category of sets (i.e. functions) that preserve the operations, relations, etc. – i.e. "homomorphisms," in the usual sense.

But there are lots of other examples where the notion of morphism is given directly, and that then determines the corresponding notion of "structure" – e.g. "differentiable structure" or "smooth structure".

# III. Objects

If two objects are isomorphic, they stay that way if we forget some of the structure.

# III. Objects

If two objects are isomorphic, they stay that way if we forget some of the structure.
For example, isomorphic groups are also isomorphic as sets,

$$G \cong H \;\Rightarrow\; |G| \cong |H|.$$

# III. Objects

If two objects are isomorphic, they stay that way if we forget some of the structure.
For example, isomorphic groups are also isomorphic as sets,

$$G \cong H \;\Rightarrow\; |G| \cong |H|.$$

Therefore, one can *distinguish* different structures by taking them to non-isomorphic underlying sets.

# III. Objects

If two objects are isomorphic, they stay that way if we forget some of the structure.
For example, isomorphic groups are also isomorphic as sets,

$$G \cong H \;\Rightarrow\; |G| \cong |H|.$$

Therefore, one can *distinguish* different structures by taking them to non-isomorphic underlying sets.
Any functor preserves isomorphisms, so we can distinguish non-isomorphic structures by taking them to non-isomorphic objects by some functor.

# III. Objects

For example, consider the two simple partial orders:


Are they isomorphic?

# III. Objects

For example, consider the two simple partial orders:


Are they isomorphic?
Without even thinking about order-preserving maps back and forth, we can simply count the number of such maps from the poset

$$I = (\cdot \longrightarrow \cdot)$$

# III. Objects

For example, consider the two simple partial orders:


Are they isomorphic?
Without even thinking about order-preserving maps back and forth, we can simply count the number of such maps from the poset

$$I = (\cdot \longrightarrow \cdot)$$

We see immediately that $A$ has 5, and $B$ has 6.

# III. Objects

For example, consider the two simple partial orders:


Are they isomorphic?
Without even thinking about order-preserving maps back and forth, we can simply count the number of such maps from the poset

$$I = (\cdot \longrightarrow \cdot)$$

We see immediately that $A$ has 5, and $B$ has 6.
Since $\mathrm{Hom}(I, -)$ is a functor, it follows immediately that $A$ and $B$ cannot be isomorphic.

# III. Objects

A property $P(X)$ like this is called an *invariant*:

$$A \cong B \ \& \ P(A) \ \Rightarrow \ P(B)$$

# III. Objects

A property $P(X)$ like this is called an *invariant*:

$$A \cong B \ \& \ P(A) \ \Rightarrow \ P(B)$$

It's a property that respects isomorphism, i.e. a "structural property".

# III. Objects

A property $P(X)$ like this is called an *invariant*:

$$A \cong B \;\&\; P(A) \;\Rightarrow\; P(B)$$

It's a property that respects isomorphism, i.e. a "structural property".
If in the (restricted) Principle of Structuralism (PS') we restrict attention to only structural properties $P$, we get the statement:

$$A \cong B \;\Leftrightarrow\; \text{for all structural properties } P,\ P(A) \Rightarrow P(B).$$

# III. Objects

A property $P(X)$ like this is called an *invariant*:

$$A \cong B \ \& \ P(A) \ \Rightarrow \ P(B)$$

It's a property that respects isomorphism, i.e. a "structural property".
If in the (restricted) Principle of Structuralism (PS') we restrict attention to only structural properties $P$, we get the statement:

$$A \cong B \ \Leftrightarrow \ \text{for all structural properties } P, \ P(A) \Rightarrow P(B).$$

That is, "isomorphic objects have all the same structural properties" (and conversely).

# III. Objects

This statement is trivially true, but it captures the way in which (PS') is actually used in practice:

# III. Objects

This statement is trivially true, but it captures the way in which (PS') is actually used in practice:

> *If* $A \cong B$ *and* $P(X)$ *is any structural property such that* $P(A)$*, then also* $P(B)$*.*

# III. Objects

This statement is trivially true, but it captures the way in which (PS') is actually used in practice:

> *If* $A \cong B$ *and* $P(X)$ *is any structural property such that* $P(A)$*, then also* $P(B)$*.*

In order for this to actually be *useful*, we need to be able to recognize that $P(X)$ is structural, without going through the trouble of proving that it respects isomorphisms!

# III. Objects

This statement is trivially true, but it captures the way in which (PS') is actually used in practice:

> *If $A \cong B$ and $P(X)$ is any structural property such that $P(A)$, then also $P(B)$.*

In order for this to actually be *useful*, we need to be able to recognize that $P(X)$ is structural, without going through the trouble of proving that it respects isomorphisms!

One way to do this was functoriality. Let us consider another common way.

# III. Objects

A large and useful class of structural properties are those that can be defined in a foundational system of *type theory*.

# III. Objects

A large and useful class of structural properties are those that can be defined in a foundational system of *type theory*.

The basic operations of *constructive type theory* are as follows:

$$0,\ 1,\ A+B,\ A\times B,\ A\to B,\ \Sigma_{x:A}B(x),\ \Pi_{x:A}B(x)$$

# III. Objects

A large and useful class of structural properties are those that can be defined in a foundational system of *type theory*.

The basic operations of *constructive type theory* are as follows:

$$0, \ 1, \ A + B, \ A \times B, \ A \to B, \ \Sigma_{x:A} B(x), \ \Pi_{x:A} B(x)$$

These correspond to the logical propositions:

$$\bot, \ \top, \ A \vee B, \ A \wedge B, \ A \Rightarrow B, \ \exists x : A. \ B(x), \ \forall x : A. \ B(x).$$

# III. Objects

A large and useful class of structural properties are those that can be defined in a foundational system of *type theory*.

The basic operations of *constructive type theory* are as follows:

$$0,\ 1,\ A+B,\ A\times B,\ A\to B,\ \Sigma_{x:A}B(x),\ \Pi_{x:A}B(x)$$

These correspond to the logical propositions:

$$\bot,\ \top,\ A\vee B,\ A\wedge B,\ A\Rightarrow B,\ \exists x:A.\ B(x),\ \forall x:A.\ B(x).$$

The correspondence is called "Curry-Howard" or "Propositions-as-Types". It is given by:

$$\text{term} : \text{Type} \quad \approx \quad \text{proof} : \text{Proposition}$$

The terms of a type can be regarded as the proofs of the corresponding proposition.

# III. Objects

The system of type theory has the important property that *every definable property of objects is invariant*.

# III. Objects

The system of type theory has the important property that *every definable property of objects is invariant*.
Thus if $P(X)$ is any type that is definable in the system, then the following inference is derivable:

$$\frac{A \cong B \qquad P(A)}{P(B)}$$

# III. Objects

The system of type theory has the important property that *every definable property of objects is invariant*.
Thus if $P(X)$ is any type that is definable in the system, then the following inference is derivable:

$$\frac{A \cong B \qquad P(A)}{P(B)}$$

The proof is a straightforward induction on the form of $P(X)$.

# III. Objects

The system of type theory has the important property that *every definable property of objects is invariant*.
Thus if $P(X)$ is any type that is definable in the system, then the following inference is derivable:

$$\frac{A \cong B \qquad P(A)}{P(B)}$$

The proof is a straightforward induction on the form of $P(X)$.
We record this as the following *Principle of Invariance* for type theory:

All definable properties are isomorphism invariant. (PI)

# IV. Identity

What is meant by "the objects $A$ and $B$ are identical"?

# IV. Identity

What is meant by "the objects $A$ and $B$ are identical"?
In set theory it means $\forall x.\ x \in A \leftrightarrow x \in B$, but we have already opted out of set theory, for good reasons.

# IV. Identity

What is meant by "the objects $A$ and $B$ are identical"?
In set theory it means $\forall x.\ x \in A \leftrightarrow x \in B$, but we have already opted out of set theory, for good reasons.
In *impredicative* type theory, one can define identity by Leibniz's Law as:

$$A = B \ := \ \forall P.\ P(A) \to P(B),$$

But this is identity between terms $A, B : X$ of some type $X$.

# IV. Identity

What is meant by "the objects $A$ and $B$ are identical"?
In set theory it means $\forall x.\ x \in A \leftrightarrow x \in B$, but we have already opted out of set theory, for good reasons.
In *impredicative* type theory, one can define identity by Leibniz's Law as:

$$A = B \ := \ \forall P.\ P(A) \to P(B),$$

But this is identity between terms $A, B : X$ of some type $X$.
So, more explicitly:

$$A =_X B \ := \ \forall P : \mathcal{P}(X).\ P(A) \to P(B),$$

which is not the relation we are looking for, i.e. the one between *types* $X$ and $Y$.

# IV. Identity

In predicative type theory, there is a primitive identity relation

$$x =_X y$$

on each type $X$, extending the Curry-Howard correspondence.

# IV. Identity

In predicative type theory, there is a primitive identity relation

$$x =_X y$$

on each type $X$, extending the Curry-Howard correspondence. Its rules allow the inference,

$$\frac{a =_X b \qquad P(a)}{P(b)}$$

for any predicate $P(x)$ on $X$.

# IV. Identity

If we add a universe $U$ of all (small) types, and then it too has an identity relation satisfying:

$$\frac{A =_U B \qquad P(A)}{P(B)}$$

for any definable property $P(X)$ of types.

# IV. Identity

Given identity, the property of being isomorphic to a fixed object $A$ is definable in type theory:

$$P(X) := A \cong X$$

# IV. Identity

Given identity, the property of being isomorphic to a fixed object $A$ is definable in type theory:

$$P(X) := A \cong X$$

Thus we get:

$$A =_U B \;\Rightarrow\; A \cong B,$$

since clearly $A \cong A$ by the identity function.

# IV. Identity

Given identity, the property of being isomorphic to a fixed object $A$ is definable in type theory:

$$P(X) := A \cong X$$

Thus we get:

$$A =_U B \;\Rightarrow\; A \cong B,$$

since clearly $A \cong A$ by the identity function.
We therefore have the converse of (PS), namely the expected statement:

Identical objects are isomorphic.

# IV. Identity

On the other hand, we just observed in (PI) that all definable properties $P(X)$ are invariant — before we added a universe.

# IV. Identity

On the other hand, we just observed in (PI) that all definable properties $P(X)$ are invariant — before we added a universe. Thus for such definable properties $P$ we know:

$$\frac{A \cong B \qquad P(A)}{P(B)}$$

# IV. Identity

On the other hand, we just observed in (PI) that all definable properties $P(X)$ are invariant — before we added a universe. Thus for such definable properties $P$ we know:

$$\frac{A \cong B \qquad P(A)}{P(B)}$$

If this held generally, for *all* definable properties $P$, we could do the same trick for identity that we just did for isomorphism:

# IV. Identity

On the other hand, we just observed in (PI) that all definable properties $P(X)$ are invariant — before we added a universe. Thus for such definable properties $P$ we know:

$$\frac{A \cong B \qquad P(A)}{P(B)}$$

If this held generally, for *all* definable properties $P$, we could do the same trick for identity that we just did for isomorphism:
Namely, taking

$$P(X) := \ A =_U X,$$

we infer as before:

$$A \cong B \ \Rightarrow \ A =_U B,$$

i.e. the Principle of Structuralism.

# IV. Identity

So, in the full system of type theory, the Principle of Invariance implies the Principle of Structuralism.

# IV. Identity

So, in the full system of type theory, the Principle of Invariance implies the Principle of Structuralism.

More simply: if in type theory with identity and a universe, it is still the case that all definable properties are isomorphism invariant, then isomorphic objects would be identical.

# IV. Identity

So, in the full system of type theory, the Principle of Invariance implies the Principle of Structuralism.

More simply: if in type theory with identity and a universe, it is still the case that all definable properties are isomorphism invariant, then isomorphic objects would be identical.

Is that really possible?

# V. What does "are" mean?

In type theory, every proposition is a type. For instance, corresponding to the proposition $A \cong B$ we have the type of isomorphisms between $A$ and $B$:

$$\text{``}A \cong B\text{''} \rightsquigarrow \mathrm{Iso}(A, B)$$

# V. What does “are” mean?

In type theory, every proposition is a type. For instance, corresponding to the proposition $A \cong B$ we have the type of isomorphisms between $A$ and $B$:

$$\text{“}A \cong B\text{”} \rightsquigarrow \mathsf{Iso}(A, B)$$

In order to prove that $A \cong B$, one constructs a term of type $\mathsf{Iso}(A, B)$, i.e. an isomorphism between $A$ and $B$.

# V. What does "are" mean?

In type theory, every proposition is a type. For instance, corresponding to the proposition $A \cong B$ we have the type of isomorphisms between $A$ and $B$:

$$\text{“}A \cong B\text{”} \rightsquigarrow \mathrm{Iso}(A, B)$$

In order to prove that $A \cong B$, one constructs a term of type $\mathrm{Iso}(A, B)$, i.e. an isomorphism between $A$ and $B$.
Similarly, corresponding to the proposition $A =_U B$ there is the *identity type*,

$$\text{“}A =_U B\text{”} \rightsquigarrow \mathrm{Id}(A, B).$$

# V. What does "are" mean?

In type theory, every proposition is a type. For instance, corresponding to the proposition $A \cong B$ we have the type of isomorphisms between $A$ and $B$:

$$\text{“}A \cong B\text{”} \;\rightsquigarrow\; \mathrm{Iso}(A, B)$$

In order to prove that $A \cong B$, one constructs a term of type $\mathrm{Iso}(A, B)$, i.e. an isomorphism between $A$ and $B$.
Similarly, corresponding to the proposition $A =_U B$ there is the *identity type*,

$$\text{“}A =_U B\text{”} \;\rightsquigarrow\; \mathrm{Id}(A, B).$$

This is the "type of proofs that $A$ is identical $B$".

# V. What does "are" mean?

In type theory, every proposition is a type. For instance, corresponding to the proposition $A \cong B$ we have the type of isomorphisms between $A$ and $B$:

$$\text{“}A \cong B\text{”} \rightsquigarrow \mathsf{Iso}(A, B)$$

In order to prove that $A \cong B$, one constructs a term of type $\mathsf{Iso}(A, B)$, i.e. an isomorphism between $A$ and $B$.
Similarly, corresponding to the proposition $A =_U B$ there is the *identity type*,

$$\text{“}A =_U B\text{”} \rightsquigarrow \mathsf{Id}(A, B).$$

This is the "type of proofs that $A$ is identical $B$".
There is also a neat geometric interpretation of this type ...

# V. What does “are” mean?

We just showed that there is a map,

$$\mathrm{Id}(A, B) \longrightarrow \mathrm{Iso}(A, B),$$

since the relation of isomorphism is reflexive.

# V. What does "are" mean?

We just showed that there is a map,

$$\mathsf{Id}(A, B) \longrightarrow \mathsf{Iso}(A, B),$$

since the relation of isomorphism is reflexive.

Voevodsky's *Univalence Axiom* says that this map is itself an isomorphism:

$$\mathsf{Id}(A, B) \ \cong \ \mathsf{Iso}(A, B).$$

# V. What does "are" mean?

We just showed that there is a map,

$$\mathsf{Id}(A, B) \longrightarrow \mathsf{Iso}(A, B),$$

since the relation of isomorphism is reflexive.

Voevodsky's *Univalence Axiom* says that this map is itself an isomorphism:

$$\mathsf{Id}(A, B) \;\cong\; \mathsf{Iso}(A, B).$$

In particular, there is then a map coming back,

$$\mathsf{Iso}(A, B) \longrightarrow \mathsf{Id}(A, B),$$

which may reasonably be read "isomorphic objects are identical".

# V. What does "are" mean?

We just showed that there is a map,

$$\mathsf{Id}(A, B) \longrightarrow \mathsf{Iso}(A, B),$$

since the relation of isomorphism is reflexive.

Voevodsky's *Univalence Axiom* says that this map is itself an isomorphism:

$$\mathsf{Id}(A, B) \ \cong \ \mathsf{Iso}(A, B).$$

In particular, there is then a map coming back,

$$\mathsf{Iso}(A, B) \longrightarrow \mathsf{Id}(A, B),$$

which may reasonably be read "isomorphic objects are identical".

This is just the Principle of Structuralism. It implies the inference we just doubted was even possible:

$$A \cong B \ \Rightarrow \ A =_U B.$$

# V. What does “are” mean?

The Univalence Axiom is not a philosophical claim, but rather a precise mathematical axiom in a new system for the foundations of mathematics.

# V. What does "are" mean?

The Univalence Axiom is not a philosophical claim, but rather a precise mathematical axiom in a new system for the foundations of mathematics.

It has a more general form, namely that *equivalence of objects is equivalent to identity*,

$$\mathrm{Eq}(A, B) \ \simeq \ \mathrm{Id}(A, B) \tag{UA}$$

where the notion of "equivalence" is a broad generalization of isomorphism that subsumes also logical equivalence, categorical equivalence, and homotopy equivalence.

# V. What does "are" mean?

Since the rules of identity permit substitution of identicals, one consequence of (UA) is the invariance schema that we called the Principle of Invariance:

$$\frac{A \cong B \qquad P(A)}{P(B)}$$

but now for *all* properties $P(X)$.

# V. What does "are" mean?

Since the rules of identity permit substitution of identicals, one consequence of (UA) is the invariance schema that we called the Principle of Invariance:

$$\frac{A \cong B \qquad P(A)}{P(B)}$$

but now for *all* properties $P(X)$.

Thus even in this extended system, it still holds that *all definable properties are structural*.

# V. What does “are” mean?

Since the rules of identity permit substitution of identicals, one consequence of (UA) is the invariance schema that we called the Principle of Invariance:

$$\frac{A \cong B \qquad P(A)}{P(B)}$$

but now for *all* properties $P(X)$.

Thus even in this extended system, it still holds that *all definable properties are structural*.
But note that this requires that we also add the Univalence Axiom when we add identity and a universe.

# V. What does "are" mean?

In sum, (UA) is a new axiom of logic that not only makes sense of, but actually implies, the Principle of Structuralism:

Isomorphic objects are identical. (PS)

# V. What does “are” mean?

In sum, (UA) is a new axiom of logic that not only makes sense of, but actually implies, the Principle of Structuralism:

$$\text{Isomorphic objects are identical.} \qquad \text{(PS)}$$

Note: Given a sequence of univalent universes, the equivalence between equivalence and identity itself becomes an identity — so a consequence of UA is that “equivalence is identity”:

$$(A \simeq B) = (A = B)$$

# V. What does "are" mean?

In sum, (UA) is a new axiom of logic that not only makes sense of, but actually implies, the Principle of Structuralism:

$$\text{Isomorphic objects are identical.} \tag{PS}$$

Note: Given a sequence of univalent universes, the equivalence between equivalence and identity itself becomes an identity — so a consequence of UA is that "equivalence is identity":

$$(A \simeq B) = (A = B)$$

This seems quite radical from a conventional point of view.
It is incompatible with classical foundations in set theory.
It is consistent with a type-theoretic foundation, and that is part of this remarkable new insight.

# VI. Background

From a foundational perspective, the Univalence Axiom is certainly radical and unexpected, but it is not entirely without historical precedent.

# VI. Background

From a foundational perspective, the Univalence Axiom is certainly radical and unexpected, but it is not entirely without historical precedent.

The first edition of *Principia Mathematica* used an intensional type theory, but the axiom of reducibility implied that every function has an extensionally equivalent predicative replacement. This spoiled the interpretation of functions as expressions and the substitutional interpretation of quantification, what may be called the "syntactic interpretation", as favored by Russell on the days when he was a constructivist.

# VI. Background

From a foundational perspective, the Univalence Axiom is certainly radical and unexpected, but it is not entirely without historical precedent.

The first edition of *Principia Mathematica* used an intensional type theory, but the axiom of reducibility implied that every function has an extensionally equivalent predicative replacement. This spoiled the interpretation of functions as expressions and the substitutional interpretation of quantification, what may be called the "syntactic interpretation", as favored by Russell on the days when he was a constructivist.

In the second edition, Russell states a new principle that he attributes to Wittgenstein: "a function can only occur in a proposition through its values". He says that this justifies an axiom of extensionality, doing at least some of the work of the axiom of reducibility.

# VI. Background

Wittgenstein has noticed that all functions that can be explicitly defined are in fact extensional, and so one can consistently add the axiom of extensionality without destroying the syntactic interpretation.

# VI. Background

Wittgenstein has noticed that all functions that can be explicitly defined are in fact extensional, and so one can consistently add the axiom of extensionality without destroying the syntactic interpretation.

This is similar to our move: the Univalence Axiom is also a very general extensionality principle; it implies the extensionality laws:

$$\begin{aligned} p \leftrightarrow q &\longrightarrow p = q \\ \forall x(fx \leftrightarrow gx) &\longrightarrow f = g \end{aligned}$$

# VI. Background

Wittgenstein has noticed that all functions that can be explicitly defined are in fact extensional, and so one can consistently add the axiom of extensionality without destroying the syntactic interpretation.

This is similar to our move: the Univalence Axiom is also a very general extensionality principle; it implies the extensionality laws:

$$\begin{aligned} p \leftrightarrow q &\longrightarrow p = q \\ \forall x(fx \leftrightarrow gx) &\longrightarrow f = g \end{aligned}$$

Since nothing in syntax violates UA, we can add it to the system and still maintain the good properties of syntax, like invariance under isomorphism.

# VI. Background

Carnap was one of the first people to show that all properties definable in the theory of types are invariant under isomorphisms. (He did so in pursuit of his ill-fated *Gabelbarkeitssatz*.)

# VI. Background

Carnap was one of the first people to show that all properties definable in the theory of types are invariant under isomorphisms. (He did so in pursuit of his ill-fated *Gabelbarkeitssatz*.)

Tarski later proposed this condition as an explication of the concept of a "logical notion". The idea was to generalize Klein's program from geometric to "arbitrary" permutations, in order achieve the most general notion of an "invariant", which would then be a *logical notion*.

# VI. Background

Carnap was one of the first people to show that all properties definable in the theory of types are invariant under isomorphisms. (He did so in pursuit of his ill-fated *Gabelbarkeitssatz*.)

Tarski later proposed this condition as an explication of the concept of a "logical notion". The idea was to generalize Klein's program from geometric to "arbitrary" permutations, in order achieve the most general notion of an "invariant", which would then be a *logical notion*.

Various people have pursued this idea further in recent times:

1. McGee: $\infty$-FOL is invariant under all iso.s,
2. Feferman: FOL w/o $=$ is invariant under all surj. homo.s,
3. Bonnay: FOL is invariant under all partial iso.s,
4. Awodey & Forssell: FOL is invariant under all continuous iso.s.

# Conclusion

The results of homotopy type theory presented here show that the entire system of constructive type theory is invariant under the much more general notion of *homotopy equivalence* – "same shape" – a much larger class of maps than Tarski's bijections.

# Conclusion

The results of homotopy type theory presented here show that the entire system of constructive type theory is invariant under the much more general notion of *homotopy equivalence* – "same shape" – a much larger class of maps than Tarski's bijections.

The Univalence Axiom takes this insight,

$$A \simeq B \ \&\ P(A) \ \Rightarrow \ P(B),$$

i.e. "all properties are invariant", and turns it into a simple and beautiful, new logical axiom:

$$(A \simeq B) \ \simeq \ (A = B).$$

So that "logical identity" is equivalent to equivalence.

# Conclusion

Finally, as a consequence of (UA), together with the very definition of structure (DS), we then have that two mathematical objects are identical iff they have the same structure:

$$\mathrm{str}(A) = \mathrm{str}(B) \;\Leftrightarrow\; A = B.$$

# Conclusion

Finally, as a consequence of (UA), together with the very definition of structure (DS), we then have that two mathematical objects are identical iff they have the same structure:

$$\mathrm{str}(A) = \mathrm{str}(B) \;\Leftrightarrow\; A = B.$$

In other words, mathematical objects *are* structures.

# Conclusion

Finally, as a consequence of (UA), together with the very definition of structure (DS), we then have that two mathematical objects are identical iff they have the same structure:

$$\mathrm{str}(A) = \mathrm{str}(B) \iff A = B.$$

In other words, mathematical objects *are* structures.
Could there be a stronger formulation of structuralism?