(Tierney-Lawvere) A **topos** is a presentable locally cartesian closed category with a **subobject classifier**
1940 A. Church *A Formulation of the Simple Theory of Types*

Extremely simple and natural

A type `bool` as a type of propositions

A type `I` for individuals

Function type `A → B`

Natural semantics of types as sets
In set theory, a function is a *functional graph*

In type theory, a function is given by an *explicit definition*

If \( t : B \), we can introduce \( f \) of type \( A \rightarrow B \) by the definition

\[
 f(x) = t
\]

\( f(a) \) «reduces» to \( (a/x)t \) if \( a \) is of type \( A \)
Functions in simple type theory

We have two notions of function

- functional graph
- function explicitly defined by a term

What is the connection between these two notions?

Church introduces a special operation $\iota x.P(x)$ and the "axiom of description". If $\exists!x : A.P(x)$ then $P(\iota x.P(x))$
We can then define a function from a functional graph

$$\forall x.\exists!y. R(x, y) \rightarrow \exists f. \forall x. R(x, f(x))$$

by taking $$f(x) = \iota y. R(x, y)$$

By contrast, Hilbert’s operation $$\epsilon x. P(x)$$ (also used by Bourbaki) satisfies

if $$\exists x : A. P(x)$$ then $$P(\epsilon x. P(x))$$

To use $$\exists!x : A. \varphi$$ presupposes a notion of equality on the type $$A$$
Rules of equality

Equality can be specified by the following purely logical rules

(1) $a =_A a$

(2) if $a_0 =_A a_1$ and $P(a_0)$ then $P(a_1)$
Equality in mathematics

The first axiom of set theory is the axiom of extensionality stating that two sets are equal if they have the same element.

In Church’s system we have two form of the axiom of extensionality

(1) two equivalent propositions are equal

\[(P \equiv Q) \rightarrow P =_{bool} Q\]

(2) two pointwise equal functions are equal

\[(\forall x : A. f(x) =_B g(x)) \rightarrow f =_{A \rightarrow B} g\]

The univalence axiom is a generalization of (1)
Logic and Topology

Limitation of simple type theory

We can form

\[ I \rightarrow \text{bool}, \ (I \rightarrow \text{bool}) \rightarrow \text{bool}, \ ((I \rightarrow \text{bool}) \rightarrow \text{bool}) \rightarrow \text{bool}, \ldots \]

but not talk *internally* about the family of such types

We cannot introduce an *arbitrary* structure (ring, group, \ldots)
The basic notion is the one of *family of types* $B(x), \ x : A$

We describe directly some *primitive* operations

$$(\prod x : A)B(x) \quad f \quad \text{where } f(x) = b$$

$$(\Sigma x : A)B(x) \quad (a, b)$$

$A + B \quad i(a), j(b)$$

which are *derived* operations in set theory
Logical operations are reduced to constructions on types by the following dictionary:

\[
\begin{align*}
A \land B & \quad A \times B = (\Sigma x : A)B \\
A \lor B & \quad A + B \\
A \rightarrow B & \quad A \rightarrow B = (\Pi x : A)B \\
(\forall x : A)B(x) & \quad (\Pi x : A)B(x) \\
(\exists x : A)B(x) & \quad (\Sigma x : A)B(x)
\end{align*}
\]
Dependent types

de Bruijn (1967) notices that this approach is suitable for representation of mathematical proofs on a computer (AUTOMATH)

Proving a proposition is reduced to building an element of a given type

“This reminds me of the very interesting language AUTOMATH, invented by N. G. de Bruijn. AUTOMATH is not a programming language, it is a language for expressing proofs of mathematical theorems. The interesting thing is that AUTOMATH works entirely by type declarations, without any need for traditional logic! I urge you to spend a couple of days looking at AUTOMATH, since it is the epitome of the concept of type.”

D. Knuth (1973, letter to Hoare)
Dependent types

Two ways of introducing dependent types

(1) Universes

(2) Path types
Universes

A universe is a type the element of which are types, and which is closed by the operations

\[(\Pi x : A)B(x) \quad (\Sigma x : A)B(x) \quad A + B\]

Russell’s paradox does not apply directly since one cannot express \(X : X\) as a type

However, Girard (1971) shows how to represent Burali-Forti paradox if one introduces a type of all types
Martin-Löf (1973), following Grothendieck, introduces a hierarchy of universes:

\[ U_0 : U_1 : U_2 : \ldots \]

Each universe \( U_n \) is closed by the operations:

\[ (\Pi x : A)B(x) \quad (\Sigma x : A)B(x) \quad A + B \]
Universes and dependent sums

We can formally represent the notion of structure

\[(\Sigma X : U_0)((X \times X \to X) \times X)\]

collection of types with a binary operation and a constant

\[(X \times X \to X) \times X\] family of types for \(X : U_0\)

This kind of representation is used by Girard for expressing Burali-Forti paradox
It is now represented by a dependent family of type $\text{Path}(A, a, b)$

We have the constant path $1_a : \text{Path}(A, a, a)$ and if $p : \text{Path}(A, a, b)$ the transport function

$C(a) \rightarrow C(b)$

which is reminiscent of the path lifting condition
Voevodsky introduced the definitions

\[
\text{isContr}(A) = (\Sigma a : A)(\Pi x : A)\text{Path}(A, a, x)
\]

\[
\text{Fiber}(f, a) = (\Sigma x : T)\text{Path}(A, f(x), a) \text{ for } f : T \to A
\]

\[
\text{isEquiv}(f) = (\Pi a : A)\text{isContr} (\text{Fiber}(f, a))
\]

\[
\text{Equiv}(T, A) = (\Sigma f : T \to A)\text{isEquiv}(f)
\]

\[
\text{isProp}(X) = (\Pi a : X)(\Pi b : X)\text{Path}(X, a, b)
\]

\[
\text{isSet}(X) = (\Pi a : X)(\Pi b : X)\text{isProp} (\text{Path}(X, a, b))
\]
Voevodsky proves for instance that

given $\psi : (\Pi a : A) \, B(a) \to C(a)$

we can define $\psi' : (\Sigma a : A)B(a) \to (\Sigma a : A)C(a)$

then

$\text{isEquiv}(\psi') \leftrightarrow (\Pi a : A)\text{isEquiv}(\psi(a))$
Martin-Löf introduced, for purely formal logical reasons, the law

\[(\Pi a : A)\text{isContr}((\Sigma x : A)\text{Path}(A, a, x))\]

This expresses that the total space of the fibration defined by the space of paths having a given origin is \textit{contractible}.

This is exactly the starting point of the loop-space method in algebraic topology (J.P. Serre)
The canonical map

\[ \text{Path}(U, A, B) \to \text{Equiv}(A, B) \]

is itself an equivalence (original statement)

This generalizes the fact that two equivalent propositions are equal!

Another (equivalent) statement is

\[(\Pi A : U) \text{isContr(}(\Sigma X : U) \text{Equiv}(A, X))\]
Equality type

\( \text{isContr}( (\Sigma x : A) B(x) ) \) is a uniform generalization of

\( (\exists! x : A) B(x) \)

and we have a description operator since

\( \text{isContr}( (\Sigma x : A) B(x) ) \rightarrow (\Sigma x : A) B(x) \)

unique existence implies effective existence
In the 50s, development of a “combinatorial” notion of spaces

D. Kan: first with cubical sets (1955) then with simplicial sets

“A combinatorial definition of homotopy groups” (1958)

These spaces form a cartesian closed category

Moore (1955)

They form a model of type theory with the univalence axiom

Voevodsky (2009)
Algebraic topology

Proof of Moore's Theorem

“assez technique et délicate” (H. Cartan, Séminaire E.N.S. 56-57)
Existence of dependent product reduces to the fact that trivial cofibrations are stable under pullbacks along Kan fibrations

The proof of which is quite complex (uses minimal fibrations?)
All these *results* are intrinsically *non effective*

If one expresses the definitions as they are in IZF then the following facts are *not* provable

(1) If $E \to B$ fibration and $b_0, b_1$ path-connected then $E(b_0)$ and $E(b_1)$ are homotopy equivalent (j.w.w. M. Bezem, 2015)

(2) Moore’s Theorem (E. Parmann, 2015)
Algebraic topology

The “reason” is that all these arguments use reasoning by case whether a complex is degenerate or not

(already in J.P. Serre’s thesis 1951)

This is not decidable in general in an effective framework

The arguments are not “uniform” and non elementary
Univalent Foundations

I will now present a possible effective combinatorial notion of spaces with higher-order notion of connectedness

This is done in a constructive setting

We can extract from this a purely syntactical type system with no axioms
For each finite set $I$ we introduce a formal representation of the cube $[0, 1]^I$.

$[0, 1]$ has a structure of *de Morgan algebra*.

Bounded distributive lattice with a de Morgan involution.

We can consider the free de Morgan algebra $dM(S)$ on any set $S$.

It is finite if $S$ is finite.

This defines a monad $dM$ on the category of finite sets.

$C$ is the opposite of the Kleisli category of $dM$. 

---

**Logic and Topology**

**Category of cubes**
A map $J \to I$ in $C$ is a set theoretic map $I \to \text{dM}(J)$

A *cubical set* if a presheaf on $C$

Family of sets $X(I)$ with transition functions

$X(I) \to X(J)$ for $f : J \to I$

$u \mapsto uf$

If $I$ finite set, we write also $I$ the representable functor it defines

So $I$ represents a cubical set
We have direct face maps \((i_0), (i_1) : I \to I, i\) that are monos.

A face map is a composition of direct face maps.

Any map \(f : J \to I\) can be uniquely decomposed

\[ f = gh \]

where \(g\) is a face map and \(h : J \to K\) is strict i.e. the corresponding map \(K \to \text{dM}(J)\) never takes the value 0 or 1.
Singular cubical sets

We have a functor $\mathcal{C} \to \text{Top}$

$I \mapsto [0, 1]^I$

Any topological space $X$ defines a singular cubical set $S(X)$

$S(X)(I)$ is the set of all continuous maps $[0, 1]^I \to X$
This defines a cubical set, which represents the interval $\mathbb{I}$ has a de Morgan algebra structure.
If $X$ cubical set, the path “space” of $X$ is $X^I$

An element of $X^I(J)$ is defined by an element of $X(J,i)$ with $i$ not in $J$

$u$ in $X(J,i)$ and $v$ in $X(J,k)$ represents the same element if, and only if,

$u(i = k) = v$ in $X(J,k)$
Homotopy between the constant path on \( a \) and any path \( p : \text{Path} \ A \ a \ b \)

\[
\begin{array}{c}
\text{a} \\
\uparrow \\
\text{a}
\end{array} 
\xrightarrow{p \ i} 
\begin{array}{c}
\text{b} \\
\uparrow \\
\text{j}
\end{array}
\]

\[
\begin{array}{c}
\text{a} \\
\uparrow \\
\text{a}
\end{array} 
\xrightarrow{p \ (i \land j)} 
\begin{array}{c}
\text{b} \\
\uparrow \\
\text{j}
\end{array} 
\xrightarrow{p \ j} 
\begin{array}{c}
\text{a} \\
\uparrow \\
\text{a}
\end{array}
\]

This expresses that \((\Sigma x : A)\text{Path}(A, a, x)\) is contractible
II has two global points 0 and 1

Ω subobject classifier

We have a map $\mathbb{I} \rightarrow \Omega$, $i \mapsto (i = 1)$

This is a lattice map

The image of this map is the face lattice $\mathbb{F} \rightarrow \Omega$

Exactly the sieves that are union of faces
Face lattice

\( F(I) \) can be defined directly as the free bounded distributive lattice generated by symbols \((i = 0), (i = 1)\) and relations

\[ 0 = (i = 0) \land (i = 1) \]

Intuitively \( F \) will classify \emph{cofibrations}

We cannot hope to have \emph{all} monos as cofibrations in an effective way
Face lattice

Any map $\psi : I \to \mathbb{F}$ determines a subpresheaf $I, \psi$ of $I$

$(I, \psi)(J)$ set of maps $f : J \to I$ such that $\psi f = 1$

$I, \psi$ is a subpresheaf of $I$

An map $g : K \to I$ determines a map $K, \psi g \to I, \psi$ that we write also $g$
A cubical set is \textit{contractible} (uniformly) if we have an operation \( \text{ext}(I, \psi, u) : I \to X \) given 
\[ \psi : I \to \mathbb{F}, \quad u : I, \psi \to X \]
\textit{such that}

(1) \( \text{ext}(I, \psi, u)f = \text{ext}(J, \psi f, uf) : J \to X \) whenever \( f : J \to I \) and

(2) \( \text{ext}(I, 1, u) = u \)

\( \text{ext}(I, \psi, u) \) is an extension of \( u \) “uniform in \( I \)”
The usual definition only requires the *existence* of an extension of a partial element \( u : I, \psi \to X \)

Here we have a “contractibility structure”

It would *not* work only to require an explicit operation without any uniformity condition.
Contractible cubical sets

$\mathbb{F}$ is a contractible cubical set

$\Omega$ is a contractible cubical set

$\mathbb{I}$ is *not* contractible

Any $u : I, \psi \to \mathbb{I}$ can be extended to $I \to \mathbb{I}$

but *not* in an uniform way! (Ch. Sattler)
Fibrant cubical sets

$X$ is fibrant if, and only if, we have an operation $\text{fill}(J, \psi, u) : J \times \mathbb{I} \to X$ given

$u : (J, \psi) \times \mathbb{I} \sqcup J \times 0 \to X$

such that

(1) $\text{fill}(J, \psi, u)(g \times 1_{\mathbb{I}}) = \text{fill}(K, \psi g, u g)$ if $g : K \to J$ and

(2) $\text{fill}(J, 1, u) = u$
Fibrant cubical sets

**Theorem:** Any singular cubical set $S(X)$ is fibrant
Cofibration-trivial fibration factorization

(A. Swan) What happens to the “small object argument”

Given $\sigma : A \to B$ where $A$ and $B$ are cubical sets

Define $C(I)$ having for element $v, \psi, u$ with

$v : I \to B$

$\psi$ in $F(I)$

$u : I, \psi \to A$ such that $v$ extends $\sigma u$

$A \to C, \quad u \longmapsto (\sigma u, 1, u)$

$C \to B, \quad (v, \psi, u) \longmapsto v$
(A. Swan) We can use this to define $\text{Id}(A, a, b)$ from $\text{Path}(A, a, b)$
The definition of $U_k$ is as usual

We consider $C = (\Sigma X : U_k)\text{Equiv}(A, X)$

We have the first projection $p : C \to U_k$

**Main algorithm:** For any partial element $u : I, \psi \to C$ find a total extension of $pu : I, \psi \to U_k$

A crucial point is the fact that we have an operation $\forall : \mathbb{F}^I \to \mathbb{F}$
From this follows

1. $U_k$ is fibrant, since any path $\text{Path}(U_k, A, B)$ defines an equivalence in $\text{Equiv}(A, B)$

2. Univalence in the form that $C = (\Sigma X : U_k)\text{Equiv}(A, X)$ is contractible
Higher-Inductive Types

We can define spheres $S^n : U_0$ inductively

Propositional truncation $\text{inh} : U_k \rightarrow U_k$
A $U_k$-torsor is a type $X : U_k$ with a $\mathbb{Z}$-action such that

1. for any $u$ in $X$ the map $n \mapsto u + n$, $\mathbb{Z} \to X$ is an equivalence
2. and $\text{inh}(X)$

If $X$ is a torsor we cannot in general exhibit one element of $X$

It follows from (1) and (2) that $X$ is a set
The collection of all torsors form a groupoid $G_k$ which is equivalent to $S^1$

This can be proved without the axiom of choice!

All types $G_k$ are equivalent to $S^1 : U_0$

It should be consistent to add $G_k : U_0$