

## 1. Set theory (start)

### 1.1. Introduction to the foundation of mathematics

Mathematics is the study of systems of elementary objects, whose existence is abstract (independent of the physical world), and the only nature of each component (element, relation) is to be exact, unambiguous: two objects are equal or different, related or not, an operation gives an exact result...

Mathematical logic is the study of the foundation of mathematics, the mathematical description of the world of mathematics itself. Mathematics is split into theories, implicit or explicit frameworks of any mathematical study. Each theory is the study of systems (or “worlds”...) of some precisely described form, called its *foundation*: its lists of types of components, of structures relating them, and of properties (formulas) called *axioms* they must satisfy. There is also a hierarchy between theories, where some can be a foundation for others. For instance, the foundations of several theories may have a common part forming a simpler theory, whose developments are applicable to all.

Mathematics, or each theory, can be understood in two ways: a realistic and a formalistic view.

The realistic view (also called *idealistic* or *platonian*) focuses on the worlds or systems studied, seen as preexisting realities to be explored (Plato called that a remembering). This is the approach of intuition that smells the global order of things.

The formalistic view is the syntactical and rigorous approach to a theory, starting from its foundation (its formal expression), and following the rules of progression.

First and at any time, the foundation of mathematics, or of a theory, is what is known or chosenly accepted of it. Then, the study progresses by choosing some possible developments (new concepts and information) resulting from the previous foundation and joining it to form the next foundation. Other options not chosen at a time can still come later, as the foundation that could generate them remains. Thus, developments are anyway the exploration of a reality defined as the totality of possible developments. A fundamental work is to develop from a simple initial basis, a more complete foundation that best eases further interesting developments.

Of course, human thought cannot operate in a fully realistic way, but only make reasonings more or less translatable into a formal development from some foundation (which prevents the errors of intuition). Moreover, the reality itself of the mathematical world exceeds any realistic conception that can be made of it: any consideration of a reality needs to be specified by some theory, but no theory can exhaust the mathematical reality that can enlarge beyond any totality once assumed. Fortunately, some rather simple theories can modelize quite sufficiently large realities for most needs.

But formalism is not absolute either, as each formalism’s clarity and elementarity is relative: it had to be chosen, defined and motivated more or less arbitrarily from elsewhere in the mathematical reality, intuitively or from some previous formalism. An intuitive vision of a problem may seem clearer than a formal argument. In practice we use semi-formal proofs, just rigorous enough for an intuition to feel that a complete formalization is possible, yet not fully written.

Despite the elementarity of mathematical objects, the foundation of mathematics (with the foundation of each foundation, unlike ordinary mathematical works going forward from an assumed foundation), turns out to be quite complex (though not as bad as a physics’theory of everything): instead of a starting point, it forms a wide dynamic almost looped to itself, with easier and harder steps.

(This is similar to dictionaries defining each word by other words, or to another science of finite systems: computer programming. Indeed computers can be simply used, knowing what you do but not why it works. Their working is based on software that was written in some language, then compiled by other software, and on the hardware and processor whose design and production were computer assisted. And this is much better than at the birth of this field.)

This loop is a true foundation for mathematics, as its parts are rich of useful concepts for diverse branches of mathematics. It is dominated by two theories:

**Set theory** studies mathematical objects, from the simplest to the most complex such as infinite systems. But there is a (potentially infinite) diversity of possible (non-equivalent) set theories.

**Model theory** is the theory of (axiomatic) theories, with their formalism (the rules of mathematical language: the structure of formulas as systems of symbols, rules of proof), and the systems (worlds), also called *models*, they may be studying (where they make sense). It is essentially unique, giving a clear and definitive meaning to the concepts of theory and theorem in each theory (it ought to be introduced at the undergraduate level, without formalizing the notions of formulas and proofs).

Each one is the right framework to formalize the other: each set theory can naturally be formalized as an axiomatic theory described by model theory, while the latter comes by developing set theory (theories and models are better built as objects in set theory, than by directly formalizing model theory as a theory). But these formalizations will require a long work to be complete.

Specialists in logics developed an axiomatic set theory named Zermelo-Fraenkel (ZF, or ZFC with the axiom of choice) to fit their need of a powerful theory in an enlarged founding cycle, that can prove many difficult propositions or their unprovability.

Mathematics start by introducing some simple concepts (in the founding cycle), which may seem to not rely on other ones. It is natural to start with a set theory called *naive*, that is not fully formalized as an axiomatic theory.

Naive set theories are usually presented as a popularized or implicit version of ZFC, accepting its axioms as necessary or obvious. But the meaning and use of axioms rely on other prerequisites (model theory), to which they aim to add further information. Also, ZFC is not an ideal reference for a naive set theory. Its axioms are less natural than they may seem; their full justification is more subtle than a mere intuition historically selected for the coherence and convenience of the resulting system. The many tools (concepts and theorems) needed for ordinary mathematics, are hard to develop from it. It assumes everything is a set, and thus a set of sets and so on, built over the empty set; but in practice, many objects are only used as pure elements. As sets can be used like elements, this use did not need to be formalized, but remained a discrepancy between the “theory” and the practice of mathematics.

## 1.2. Variables, sets, functions and operations

We shall start mathematics by a better suited set theory (directly in line with the practice of mathematics), starting with 3 notions (sorts of objects) : elements, sets and functions. Its formalism will be progressively developed as needed, with other notions (that can either be seen as primitive, or as defined from the former), symbols and axioms (sometimes optional). More concepts and explanations on the context of foundations (model theory) and its main subtleties (paradoxes) will complete this.

### *Constants*

A *constant symbol* is a symbol denoting a unique object, called its *value*. Examples: 3,  $\emptyset$ ,  $\mathbb{N}$ . Those of English language are proper names and names with “the” (in singular without complement).

### *Free and bound variables*

A *variable symbol* (or simply a *variable*), is a symbol without a fixed interpretation. Each possible interpretation gives it a particular value and thus sees it as a constant.

It can be imagined as limited by a box. Inside, it is usable as a constant (keeping the same value): it is called *free* or *fixed*. Outside, its possible values are perceived globally: it is called *bound*.

### *Ranges and sets*

The *range* of a variable, is its meaning when seen as bound: it is the knowledge (specification) of all its possible or authorized values called the *elements* of this range. Any range of a variable is called a *set*. (This “knowledge” is an abstract entity that can encompass infinities of objects, unlike human thought; elements are seen in bulk: unordered, ignoring their context). A variable has a range if it can be bound, i.e. when an encompassing view over all its possible values is available.

A variable is said to *browse* a set, if it is bound with this set as range. Any number of variables can be introduced with a given range, independently of each other and of other variables.

Cantor defined a set as “a grouping into a whole of distinct objects of our intuition or our thought”. He explained to Dedekind : “If the totality of elements of a multiplicity can be thought of as “simultaneously existing”, so that it can be conceived as a “single object” (or “completed object”), I call it a consistent multiplicity or a “set”.” (We expressed this “multiplicity” as that of values of a variable).

He described the opposite case as an “inconsistent multiplicity” where “admitting a co-existence of all its elements leads to a contradiction”. But non-contradiction cannot suffice to generally define sets: it is often itself unprovable (by the incompleteness of mathematics, see the “metamathematical complements” sections); non-contradiction does not mean truth (not all truths are provable); and two separately consistent coexistences might contradict each other (Omnipotence paradox).

Systematically renaming a bound variable in all its box, into another symbol not used in the same context (same box), with the same range, does not change the meaning of the whole. In practice, the same letter can designate several separate bound variables (with separate boxes), that can take different values, as they are never interpreted (free) together and are therefore incomparable. The common language does this continuously, using very few variable symbols (“he”, “she”, “it”...)

## Functions

A *function* is any object  $f$  behaving as a variable depending on another variable with range noted  $\text{Dom } f$ , called its *argument*: whenever this argument is fixed (noted as a symbol  $x$ ),  $f$  becomes a constant (noted  $f(x)$ ). In other words,  $f$  is made of the following data:

- A set called the *domain* of  $f$ , noted  $\text{Dom } f$
- For each element  $x$  of  $\text{Dom } f$ , an object  $f(x)$  called the *image of  $x$  by  $f$*  or *value of  $f$  at  $x$* .

## Operations

An *operation* is a generalized function to the case of a finite list of arguments (variables with given respective ranges), giving a result (a value) when all arguments are fixed. The number  $n$  of arguments is called its *arity*; the operation is called  *$n$ -ary*. It is called *unary* if  $n = 1$  (it is a function), *binary* if  $n = 2$ , *ternary* if  $n = 3$ . . . Nullary operations are useless as replaceable by their value; we shall see how to construct those of arity  $> 1$  by means of functions.

The value of a binary operation  $f$  in its fixed arguments named (with values given by)  $x$  and  $y$ , is denoted  $f(x, y)$ . So instead of symbols, the arguments are represented by the left and right spaces in parenthesis, to be filled by any expression giving them desired values.

### 1.3. Structure of theories: objects, meta-objects, types and structures

We will first discuss several theories and models, thus be in model theory. Then focus on one theory (of sets) with a supposedly fixed model: the variables “theory” and “model” once fixed, we will be in *one-model theory*.

Each theory has its own *notions*, commonly named by common names, representing in each model the possible ranges of its variables. Model theory includes the notions of theory and model, that disappear in one-model theory because of their fixation.

Any model  $M$  of a theory  $T$  gives a model  $M_1 = [T, M]$  to the one-model theory  $T_1$ . As the common names of the notions of  $T_1$  normally express their interpretation in  $M_1$ , the prefix meta- will be added to express their interpretation in the other model  $[T_1, M_1]$  of  $T_1$ .

An *object* (of a theory in a model), is any element of a notion (possible value of a variable). By this notion of object, the one-model theory distinguishes the objects of  $T$  in  $M$  among its own objects in  $M_1$  (the meta-objects). Every object is thus a meta-object, but in practice we will only call *meta-object* those which are not objects. Unlike set theory, one-model theory gives every variable (of its studied theory) a range among notions, but keeps it as a meta-object.

A *generic theory* will be any theory expressed in a standard format (described by a more precise one-model theory, we shall specify later), that our theories can be translated to. This format clarifies the management of variables by not admitting other notions than *types* (usually in finite number) classifying both variables and objects: each object will belong to only one type, the one of variables that can name it.

Examples of notions from various theories:

Theory	Objects (notions)
Generic theory	Pure elements classified by types
Set theory	Elements, sets, functions (and possibly: operations, relations, tuples . . .)
Model Theory	Generic theories, models and their components
One-Model theory	Objects, symbols, notions, expressions, structures, axioms. . .
Arithmetic	Integers
Linear Algebra	Vectors, scalars. . .
Geometry	Points, lines, circles. . .

Notions of one-model theory classify the components of the theory (abstract types, symbols, terms, formulas), and those of the model. Symbols of generic theories are those of variables, structures (operators, predicates), connectives, and quantifiers ( $\forall$ ,  $\exists$ ). Each generic theory is distinguished by its lists of (abstract) types, of structure symbols (forming its *language*), and of axioms.

The other possible foundation of any generic theory (without one-model theory) is by translation into set theory. This integrates all theories (arithmetic, geometry. . .) and their models, in the same set theory with a fixed model (also called *universe*). But only the components of the model (and the model itself) are translated to objects; those of the theory are translated to components of set theory.

Let us describe the one-model theory notions (the form of a generic theory and its model) by such a translation that works for any generic theory: all objects will be translated into pure elements (unlike the usual view of geometry defining lines as sets of points).

The translation keeps variable symbols as such. Abstract types and structure symbols are all translated to free variables, which remain free as long as the model remains fixed. Their values are the main components of the model, those that determine it. The values of abstract types are sets of objects, called *interpreted types*.

### Structures

Each operator symbol denotes an operation called an *operator*, where each argument has a type as range, and the values are of one same type; the arity (or the list of arguments presented as spaces), the abstract type of each argument and the type of values are specified with the symbol. A nullary operator is a constant symbol, directly naming a fixed object. A unary operator, representing a function, will be called a *functor*.

The model is completed by a standard special type: that of *Booleans*, made of both elements “true” (1) and “false” (0). A variable of this type (outside the theory) is called a *Boolean variable*. Its addition to the list of types generalizes operators into *para-operators*. A *connective* is a para-operator with only Boolean arguments and values. A *predicate* is a para-operator with one or more arguments but no Boolean one, and with Boolean values. A *structure* is an operator or a predicate.

Each theory admits one symbol of equality per type (predicate with 2 arguments of this type) abusively all noted  $=$ , interpreted in the standard way. The other structures, named by the other structure symbols (arbitrary data of each theory and its model), are what makes the elements of each type play their role of possibly complex objects.

In set theory, the binary predicate  $\in$  gives sets their role: for any element  $x$  and any set  $E$ , we note  $x \in E$  ( $x$  is in  $E$ , or an element of  $E$ ), when  $x$  is a possible value of a variable with range  $E$ .

Functions play their role by the functor *Dom* (giving their domains) and the *function evaluator*, binary operator giving the value  $f(x)$  of any function  $f$  at any element  $x$  of *Dom*  $f$ .

ZFC set theory is defined as a generic theory with one type “set”, the symbols  $\in$  and  $=$ , and axioms. But our set theory will not be generic, and will include other symbols contributing to give their roles to sets, functions and other types of objects.

A one-model theory formalized as a generic theory would give types the double role of abstract and interpreted types. Also, the same meta-objects could play the roles of structure symbols and structures (thanks to the meta-structures), except that the notion of structure will include other operations between types than those named by symbols (we can not admit them “all” independently of an outside universe they would be from, irrelevant here).

## 1.4. Terms and formulas; connectives

For any theory, a *term* is a finite system of occurrences of symbols, aiming to mean an object (its value) once the model and the values of free variables are fixed. (An occurrence of a symbol is its appearance somewhere, for example “ $x + x$ ” has two occurrences of  $x$  and one of  $+$ ). We call *formula* a system similar to a term but with boolean values. An *expression* will be a term or a formula.

Expressions are built successively, each as the data (occurrence) of a symbol called its *main symbol*, and a list of previously built expressions (and possibly of variable symbols). This main symbol determines the type of values (and thus distinguishes between terms and formulas) and the format of this list (the number of entries and the type of each). For a symbol of para-operator, this format is given by the list of its arguments.

The first terms are those reduced to the occurrence of a symbol of constant or variable, called *atomic terms*. The Boolean constants 1 and 0 (nullary connectives) are the first formulas.

Other symbols require a nonempty list, thus need a convention for presenting it. Most symbols of binary para-operators appear as a character between the two arguments (the addition is noted  $x + y$  instead of the original format  $+(x, y)$  of operations); others (multiplication, power) occur implicitly, without character. Other symbols are denoted by several characters delimiting the entries. The brackets can either compose the notation of symbols like the function evaluator, or be used to distinguish the sub-expressions and their respective main symbols, as in  $(x + y)^n$ .

### Connectives

Here are the most useful connectives ordered by arity  $n > 0$ , with properties for all values of the Boolean variables (replaceable by formulas)  $A, B, C$ .

$n = 1$  :  $\neg$  (“not”) exchanges Booleans:  $\neg(1) \Leftrightarrow 0, \neg(0) \Leftrightarrow 1, \neg(\neg A) \Leftrightarrow A$ . Also denoted by barring the main symbol of its argument (forming another symbol with the same format):  $(x \neq y) \Leftrightarrow \neg(x = y)$  (“ $x$  is different from  $y$ ”);  $x \notin E \Leftrightarrow \neg(x \in E)$  (“ $x$  does not belong to  $E$ ”).

$n = 2$  :

$\wedge$  (and): is true only when both arguments are true;

$\vee$  (or): is true except when both arguments are false;

$\Rightarrow$  :  $A \Rightarrow B$  can be read “A implies B”, “A is a sufficient condition for B”, or “B is a necessary condition for A”, and means  $((\neg A) \vee B)$ . Being true except when A is true and B is false, it expresses that if A is true then B also, but otherwise it does not inform us (being true).

The formula  $(\neg B \Rightarrow \neg A)$  is called the *contrapositive* of  $(A \Rightarrow B)$ , and is equivalent to it.

$\Leftrightarrow$  (equivalent), equality of Booleans:  $(A \Leftrightarrow B) \Leftrightarrow ((A \Rightarrow B) \wedge (B \Rightarrow A))$ . A proof of  $A \Leftrightarrow B$  can be made of a proof of the implication  $(A \Rightarrow B)$ , then of the other  $(B \Rightarrow A)$ , called its *converse*.

Its negation  $A \not\Leftrightarrow B$  can also be read as negating an argument  $(A \Leftrightarrow \neg B)$ , or as “exclusive or”  $((A \vee B) \wedge \neg(A \wedge B))$ . Negations exchange various connectives:

$$(A \vee B) \not\Leftrightarrow (\neg A \wedge \neg B)$$

$$(A \wedge B) \not\Leftrightarrow (\neg A \vee \neg B)$$

$$(A \Rightarrow B) \not\Leftrightarrow (A \wedge \neg B)$$

This transforms the properties of associativity and distributivity into various formulas:

$$((A \wedge B) \wedge C) \Leftrightarrow (A \wedge (B \wedge C))$$

$$((A \vee B) \vee C) \Leftrightarrow (A \vee (B \vee C))$$

$$(A \Rightarrow (B \Rightarrow C)) \Leftrightarrow ((A \wedge B) \Rightarrow C)$$

$$(A \Rightarrow (B \vee C)) \Leftrightarrow ((A \Rightarrow B) \vee C)$$

$$(A \wedge (B \vee C)) \Leftrightarrow ((A \wedge B) \vee (A \wedge C))$$

$$(A \vee (B \wedge C)) \Leftrightarrow ((A \vee B) \wedge (A \vee C))$$

$$(A \Rightarrow (B \wedge C)) \Leftrightarrow ((A \Rightarrow B) \wedge (A \Rightarrow C))$$

$$((A \vee B) \Rightarrow C) \Leftrightarrow ((A \Rightarrow C) \wedge (B \Rightarrow C))$$

$$((A \Rightarrow B) \Rightarrow C) \Leftrightarrow ((A \vee C) \wedge (B \Rightarrow C))$$

$$(A \wedge (B \Rightarrow C)) \Leftrightarrow ((A \Rightarrow B) \Rightarrow (A \wedge C))$$

$n \geq 3$  : strings of  $\wedge$  (conjunction) such as  $(A \wedge B \wedge C)$ , and of  $\vee$  (disjunction) such as  $(A \vee B \vee C)$  are obtained by erasing the brackets thanks to associativity. The assertion of a conjunction amounts to asserting all its components. In addition, any string of formulas linked by  $\Leftrightarrow$  and  $\Rightarrow$  shorten the succession of claims (linked by  $\wedge$ ) of each link between neighboring formulas:

$$0 \Rightarrow A \Rightarrow A \Rightarrow 1$$

$$(\neg A) \Leftrightarrow (A \Rightarrow 0) \Leftrightarrow (A \Leftrightarrow 0)$$

$$(A \Rightarrow B \Rightarrow C) \Leftrightarrow ((A \Rightarrow B) \wedge (B \Rightarrow C)) \Rightarrow (A \Rightarrow C)$$

$$(A \Leftrightarrow B \Leftrightarrow C) \Leftrightarrow ((A \Leftrightarrow B) \wedge (B \Leftrightarrow C)) \Rightarrow (A \Leftrightarrow C)$$

$$(A \wedge A) \Leftrightarrow (A \wedge 1) \Leftrightarrow A \Leftrightarrow (A \vee A) \Leftrightarrow (A \vee 0) \Leftrightarrow (1 \Rightarrow A) \Leftrightarrow (A \Leftrightarrow 1)$$

$$(B \wedge A) \Leftrightarrow (A \wedge B) \Leftrightarrow (A \wedge (A \Rightarrow B)) \Rightarrow B \Rightarrow (A \vee B) \Leftrightarrow (B \vee A)$$

### *Structures defined by expressions*

In one-model theory, a *structure* (operator or predicate) will be any operation between types defined by an expression (term or formula) and a list of free variables we choose to bind in the role of arguments. This structure depends on the other variables remaining free, called its *parameters* (while the bound variables inside the expression, invisible outside, are forgotten here).

Extra symbols of structure can be introduced beyond the language of the theory, generalizing to non-zero arity the case of free variables to distinguish from constants. Such a symbol can be used to shorten the expression defining a desired structure; or used without definition in a way applicable to any structure or even more (undefinable operation). But the theory can bind a variation of such a symbol, only by binding the parameters of the expression (or a finite number of expressions) that defines the wanted values (structures). It can not encompass all structures of a given arity.

A structure is said to be *invariant* if it is defined without parameters. Any structure designated by a symbol of the language is invariant. Any addition of an invariant structure to the language

(and its definition as an axiom), is an internal development of the theory, which preserves the useful meta-notions (structure, invariant structure, provability).

#### *Axioms of equality*

For every functor  $T$  and objects  $x, y$  (that can shorten terms with parameters),

$$x = y \Rightarrow T(x) = T(y)$$

Similarly for any unary predicate  $A$  we have  $x = y \Rightarrow (A(x) \Leftrightarrow A(y))$ .

Thus an equality between terms  $x = y$  allows to replace any occurrence(s) of  $x$  by  $y$  in any expression without affecting the result. This applies in particular when a symbol is defined by a term: both are equal, and interchangeable in any expression. Also the axioms and other laws are expressed using variable symbols replaceable by terms in their use.

### 1.5. Classes in set theory

A *class* is a unary predicate seen as the set of objects where it is true. Invariant classes are very general cases of notions. They will replace types in set theory. Indeed all objects need to be included as “elements” which can belong to sets (to avoid illimited divisions between sets of elements, of sets, of functions, mixed sets. . .). Thus, the notions of sets and functions will be classes named by symbols: Set = “is a set”, Fnc = “is a function”.

(Another method would be to keep 3 types where each set would be included twice, as a set and an element, identified by a functor converting sets into elements; and the same for functions. But this idea which could be used for other theories, will not suffice in set theory which will anyway need classes as notions)

#### *Set theory in the founding cycle*

One-model theory is only partially formalisable as a theory: its requirement that each expression and each proof be of finite size, can only be expressed in its translated form into set theory. This form designating the components of its model  $M_1 = [T, M]$  by free variables, their variability makes this the set-theoretical expression of model theory (grand tour of the foundations of mathematics).

In this framework, the component  $M$  of  $M_1$  is a model of the set theoretical version of  $T$  (the set theoretical components of  $T$  are translated to terms designating their image in the system-object  $T$ ).

Now as a theory  $T$  we will formalize a set theory, interpreting its concepts in  $M$ . Therefore the meta- prefix will mark our previous interpretation of set theoretical concepts in the external universe, framework of the one-model theory concepts and even of their meta-interpretation already mentioned.

The translation of set theoretical objects, concepts and structures to their meta-set counterpart, will mainly preserve them, but is not reversible. Unlike the standard method translating all objects to pure elements, sets will usually be seen as meta-sets of the same elements (and similarly for functions). Thus any set  $E$  is a class (that of  $x$  such that  $x \in E$ , defined by this formula with argument  $x$  and parameter  $E$ ), but it will turn out that the universe, class (true) of all objects, is not a set. Any class is a meta-set of objects, but some meta-sets of objects, being undefinable, are not classes.

#### *Validity classes*

Given a model and values of the free variables, an expression will be called *valid* if it actually has a value (an invalid formula is neither true nor false). In generic theories, the validity of expressions is guaranteed by the syntactic correction, integrated into the concept of expression. But in set theory the validity of a structure (expression) may depend on the values of its arguments (free variables). The validity condition is a predicate, expressed by a formula with the same free variables, always valid. If it is unary, it defines a class that is the domain of the considered functor (or unary predicate).

Generally, the place of truth of an  $n$ -ary predicate  $\mathcal{R}$  everywhere valid, is an  *$n$ -ary class* (class of systems of values of  $n$  variables – this will be detailed later with tuples), or *multiclass*, to not specify  $n$ . It is the range of a system of  $n$  variables subjected to the hypothesis  $\mathcal{R}$ .

Any structure (defined by an expression) has for domain a multiclass, we shall thus call its *validity multiclass*, defined by the formula of its condition of validity, with the same arguments and parameters. That of  $(x \in E)$  is  $\text{Set}(E)$ ; that of the function evaluator  $f(x)$  is  $(\text{Fnc } f \wedge x \in \text{Dom } f)$ . Care must be taken to only use the expressions in their validity multiclass, which will be done rather naturally.

A theory with partially valid structures can be translated to a generic theory with one type (with structures everywhere valid), keeping intact all expressions and their values wherever they are valid: models are translated one way by adding arbitrary values to structures outside their multiclass validity (eg a constant value), and in the way back by ignoring those values.

### Extended validity

For all predicates  $A$  and  $B$  with validity conditions  $V_A$  and  $V_B$ , formulas “ $A \wedge B$ ” and “ $A \Rightarrow B$ ” will have the same validity condition ( $V_A \wedge (A \Rightarrow V_B)$ ) (breaking, for “and”, the symmetry between  $A$  and  $B$ , that need not be restored). They will thus be seen valid if  $A$  is false and  $B$  is not valid, then with respective values false and true, as obtained independently of a value arbitrarily added to  $B$ .

This makes “ $V_A \wedge A$ ” and “ $V_A \Rightarrow A$ ” always valid (respectively extending  $A$  by false and true outside its validity domain), as well as the validity conditions themselves. Formulas “ $A \wedge (B \wedge C)$ ” and “ $(A \wedge B) \wedge C$ ” have the same validity condition ( $V_A \wedge (A \Rightarrow (V_B \wedge (B \Rightarrow V_C)))$ ).

If  $(A \wedge B)$  is everywhere valid ( $A$  is everywhere valid and  $B$  valid on all the (multi)class  $A$ ), the (multi) class it defines is called the *subclass of  $A$  defined by  $B$* .

### 1.6. Bound variables in set theory

The last kind of symbol forming expressions is the binders (binding symbols), which bind one (or more) variable(s) (say  $x$ ), separating the inside (the sub-expression, using  $x$  as free) from the outside (where  $x$  is bound). Such a symbol gives a value depending on the structure, with argument  $x$ , defined by the subexpression to which it applies. This value can not accurately describe the whole structure (which is not an object), but extracts information from it.

Their format differs between set theory and generic theories, which manage ranges differently: contrary to generic theories (where ranges being types are implicit data of binding symbols), the range of a bound variable in set theory is an object (set), given as an argument to the binder (a place to fill by a term naming a set), in addition to the data of the symbol  $x$  and the target expression (the only one using  $x$ ). An expression is called *closed* if all its variables are bound.

Let us review the main binders in set theory.

#### Definitions of functions by terms

The *function definer* ( $\ni \mapsto$ ) binds a variable  $x$  with range  $E$  on a term  $t$  (depending on  $x$ ), following the syntax ( $E \ni x \mapsto t(x)$ ), sometimes shortened as ( $x \mapsto t(x)$ ) when  $E$  is determined by the context. Valid if  $t$  is valid for all  $x$  in  $E$ , it translates the functor defined by  $t$  in  $E$  (with validity domain reduced to  $E$ ) to a function with domain  $E$ . This roughly reverses the translation by the function evaluator, from functions to functors valid on sets.

Other notions will be defined as classes of objects with tools translating each object into its role and vice versa. Objects already present (sets or functions) can play the new roles, offering their tools to the new objects. The only necessary translations between objects playing the same role, will link various useful representations of a new object by old ones.

#### Formalization of operations and currying

The  $n$ -ary operations acting as  $n$ -ary operators between  $n$  sets, are formalized by:

- $n$  domain functors (of little practical use);
- an  $(n + 1)$ -ary evaluator  $f(x_1, \dots, x_n)$  with arguments  $f$  and its arguments  $x_1, \dots, x_n$ ;
- a definer, binding  $n$  variables on a term. The one binding  $x$  and  $y$  to  $E$  and  $F$  on  $t$  is denoted by  $(E \ni x, F \ni y \mapsto t(x, y))$ . The term  $(E \ni x, E \ni y \mapsto t(x, y))$  is shortened into  $(E \ni x, y \mapsto t(x, y))$ .

A way to represent operations as a class of functions, called *currying*, is to use as a definer binding  $n$  variables, the succession of  $n$  uses of the function definer (one for each variable to bind), and thus similarly as an evaluator,  $n$  uses of the function evaluator:

$$f = (E \ni x, F \ni y \mapsto t(x, y)) \simeq (E \ni x \mapsto (F \ni y \mapsto t(x, y))) = g$$

$$f(x, y) = g(x)(y) = t(x, y)$$

passing through the function  $g(x) = (F \ni y \mapsto t(x, y))$  with argument  $y$ , seeing  $x$  as free and  $y$  as bound. But this breaks the symmetry between arguments and loses the data of  $F$  when  $E$  is empty. A formalization without these flaws will be possible with the introduction of tuples (text 2).

#### Relations and set-builder symbol

A *relation* is like an operation but with Boolean values, acting as a predicate whose validity (domains of its arguments) is reduced to sets. The  $n$ -ary relations are formalized by an  $(n + 1)$ -ary predicate of evaluation, and a definer binding  $n$  variables on a formula. They could be built as operations by translating Booleans into objects. But here is the other method.

For any unary predicate  $\mathcal{R}$  valid in a set  $E$ , the subclass of  $E$  defined by  $\mathcal{R}$  is a set (range of a variable  $x$  browsing  $E$ , so that it can be bound, satisfying  $\mathcal{R}(x)$ ), denoted  $\{x \in E | \mathcal{R}(x)\}$  (set of  $x$  in  $E$  such that  $\mathcal{R}(x)$ ), called a part or subset of  $E$ : for all  $y$ ,

$$y \in \{x \in E | \mathcal{R}(x)\} \Leftrightarrow (y \in E \wedge \mathcal{R}(y))$$

The *set-builder*  $\{ \in \mid \}$ , binding  $x$  to  $E$  on  $\mathcal{R}$ , will serve to define unary relations on  $E$  as subsets  $F$  of  $E$ , evaluated by  $\in$  as predicates ( $x \in F$ ) with argument  $x$ . But these predicates are valid throughout the universe, giving 0 out of  $E$  whose data is lost. This lack of operator Dom does not matter, as the domain  $E$  is usually determined by the context.

As the function definer and set-builder symbols record the whole structure defined by the given expression on the given set, they suffice to define all other binding symbols as structures applied to them;  $n$ -ary relations can be defined by one set-builder and  $n - 1$  function definers.

More generally, for any kind of meta-objects indirectly usable in expressions as objects (a class as a set...), set theory will be enriched with tools to actualize and designate them directly as objects.

### 1.7. Quantifiers

A *quantifier* is a binding symbol on a formula, with Boolean values. A quantifier  $Q$  with domain a set  $E$  on a formula  $\mathcal{R}$  will be written  $(Qx \in E, \mathcal{R}(x))$ . If the domain is implicit (fixed), it will be written  $Qx, \mathcal{R}(x)$ . The two main quantifiers  $\forall$  and  $\exists$  (by which others will be defined later) are:

- The *existential quantifier*  $\exists$ , that reads “There exists  $x$  (in...) such that...”
- The *universal quantifier*  $\forall$ , that reads “For all (or: for any)  $x$  (in...),...”

They can be meta-defined through the meta-function  $(x \mapsto \mathcal{R}(x))$  on the same domain, by

$$(\forall x, \mathcal{R}(x)) \Leftrightarrow (x \mapsto \mathcal{R}(x)) = (x \mapsto 1) \not\Leftrightarrow (\exists x, \neg \mathcal{R}(x)) \Leftrightarrow (x \mapsto \neg \mathcal{R}(x)) \neq (x \mapsto 0)$$

In set theory,  $(\forall x \in E, \mathcal{R}(x)) \Leftrightarrow \{x \in E \mid \mathcal{R}(x)\} = E$ . The formula  $(\forall x, 1)$  is always true.

In generic theories the domain is put as an index:  $Q_i x, \mathcal{R}(x)$  where  $i$  is the type of  $x$ ; with classes,

$$\begin{aligned} (\exists_{\mathcal{C}} x, \mathcal{R}(x)) &\Leftrightarrow (\exists x, \mathcal{C}(x) \wedge \mathcal{R}(x)) \Leftrightarrow \exists_{\mathcal{C} \wedge \mathcal{R}}, 1 \\ (\forall_{\mathcal{C}} x, \mathcal{R}(x)) &\Leftrightarrow (\forall x, \mathcal{C}(x) \Rightarrow \mathcal{R}(x)) \\ \forall x, (\mathcal{C}(x) &\Leftrightarrow (\exists_{\mathcal{C}} y, x = y)) \end{aligned}$$

#### *Inclusion between classes*

A class  $\mathcal{A}$  is said to be included in a class  $\mathcal{B}$  when  $\forall x, \mathcal{A}(x) \Rightarrow \mathcal{B}(x)$ . Then  $\mathcal{A}$  is a subclass of  $\mathcal{B}$ , as  $\forall x, (\mathcal{A}(x) \Leftrightarrow (\mathcal{B}(x) \wedge \mathcal{A}(x)))$ . Conversely, any subclass of  $\mathcal{B}$  is included in  $\mathcal{B}$ . The inclusion of  $\mathcal{A}$  in  $\mathcal{B}$  implies for any predicate  $\mathcal{C}$  (in cases of validity):

$$\begin{aligned} (\forall_{\mathcal{B}} x, \mathcal{C}(x)) &\Rightarrow (\forall_{\mathcal{A}} x, \mathcal{C}(x)) \\ (\exists_{\mathcal{A}} x, \mathcal{C}(x)) &\Rightarrow (\exists_{\mathcal{B}} x, \mathcal{C}(x)) \\ (\exists_{\mathcal{C}} x, \mathcal{A}(x)) &\Rightarrow (\exists_{\mathcal{C}} x, \mathcal{B}(x)) \\ (\forall_{\mathcal{C}} x, \mathcal{A}(x)) &\Rightarrow (\forall_{\mathcal{C}} x, \mathcal{B}(x)) \end{aligned}$$

#### *Rules of proofs of quantifiers on a unary predicate $\mathcal{R}$*

**Existential Introduction.** *If we find a term  $t$  and a proof of  $\mathcal{R}(t)$ , then  $\exists x, \mathcal{R}(x)$ .*

**Existential Elimination.** *If  $\exists x, \mathcal{R}(x)$ , then we can introduce a new free variable  $z$  with the hypothesis  $\mathcal{R}(z)$  (the consequences will be true without restricting the generality).*

These rules express the meaning of  $\exists$ : going from  $t$  to  $\exists$  then from  $\exists$  to  $z$ , is like abbreviating  $t$  by  $z$ . They give the same meaning to  $\exists_{\mathcal{C}}$  as to its 2 above equivalent formulas, bypassing the extended validity rule for  $(\mathcal{C} \wedge \mathcal{R})$  by focusing on the case when  $\mathcal{C}(x)$  is true and thus  $\mathcal{R}(x)$  is valid.

While  $\exists$  appeared as the designation of an objet,  $\forall$  appears as a deduction rule:

**Universal Introduction.** *If from the mere hypothesis  $\mathcal{C}(x)$  on a new free variable  $x$  we could deduce  $\mathcal{R}(x)$ , then  $\forall_{\mathcal{C}} x, \mathcal{R}(x)$ .*

**Universal Elimination.** *If  $\forall_{\mathcal{C}} x, \mathcal{R}(x)$  and  $t$  is a term satisfying  $\mathcal{C}(t)$ , then  $\mathcal{R}(t)$  is true.*

The successive use of these rules is like rewriting the initial proof with  $x$  replaced by  $t$ .

#### *Status of open quantifiers in set theory*

Set theory is translated to a generic theory by converting to classes the domains of quantifiers:

$$\begin{aligned} (\exists x \in E, \mathcal{R}(x)) &\rightarrow (\exists x, x \in E \wedge \mathcal{R}(x)) \\ (\forall x \in E, \mathcal{R}(x)) &\rightarrow (\forall x, x \in E \Rightarrow \mathcal{R}(x)) \end{aligned}$$

Set theory only admits quantifiers over sets, called *bounded quantifiers*, in its formulas that to insist we shall call *bounded formulas* (that define its predicates and multi-classes and appear in some terms). But its translated form as a generic theory allows more general quantifiers on classes (or the universe), called *open quantifiers*. Formulas with open quantifiers will be called *claims*. The use of closed valid claims in set theory will be reserved for the declaration of their truth. These will first be axioms, then theorems (deduced from axioms), named according to their importance: a theorem is more important than a proposition; a lemma is used to prove a theorem; a corollary is deduced from one.

Except for convenience, open quantifiers will be expressed verbally in claims, articulating bounded formulas written in symbols. The above rules of proofs articulate claims with management of formulas.

The set-builder was defined by a claim. It will be used to show that the class of all sets is not a set, forcing all these distinctions between classes and sets, and between open and bounded quantifiers:

**Proposition (Russell paradox).** *For any set  $E$  there is a set  $F$  such that  $F \notin E$ .*

Proof.  $F = \{x \in E \mid \text{Set}(x) \wedge x \notin x\} \Rightarrow (F \in F \Leftrightarrow (F \in E \wedge F \notin F)) \Leftrightarrow (F \notin F \wedge F \notin E)$ .  $\square$

### 1.8. First axioms of set theory

The inclusion predicate  $\subset$  between two sets  $E$  and  $F$ , is defined by  $E \subset F \Leftrightarrow (\forall x \in E, x \in F)$ . It reads:  $E$  is included in  $F$ , or  $E$  is a part or a subset of  $F$ , or  $F$  includes  $E$ .

We always have  $E \subset E$ . Implications chains translate into inclusion chains:

$$(E \subset F \subset G) \Leftrightarrow (E \subset F \wedge F \subset G) \Rightarrow E \subset G.$$

#### First axioms:

$$\forall x, \quad \neg(\text{Set}(x) \wedge \text{Fnc}(x))$$

$$\forall_{\text{Fnc}} x, \quad \text{Set}(\text{Dom } x)$$

For any term  $t$ ,  $\forall E, \forall(\text{parameters}), \text{Fnc}(E \ni x \mapsto t(x))$

$$\forall_{\text{Set}} E, F, \quad E \subset F \subset E \Rightarrow E = F \text{ (Axiom of Extensionality).}$$

The latter redefines equality between sets by their equivalence as classes (letting elements in bulk):  $E \subset F \subset E$  means that  $E$  and  $F$  have the same elements ( $\forall x, x \in E \Leftrightarrow x \in F$ ) and implies for any predicate  $\mathcal{R}$  that  $(\forall x \in F, \mathcal{R}(x)) \Leftrightarrow (\forall x \in E, \mathcal{R}(x))$ , and similarly for  $\exists$ .

#### Translating the definer

The function definer translates to generic theory, as an infinity of symbols: for each term  $t$  with one argument (and parameters), the whole expression  $(E \ni x \mapsto t)$  is translated as a different operator symbol with arguments  $E$  and the parameters of  $t$ . (Those where every subexpression of  $t$  without any occurrence of  $x$  is the only occurrence of a parameter, suffice to define others).

The following axioms can be read as axioms of the generic theory that set theory translate into; those depending on a term  $t$  are schemas of axioms (schema of claims = infinite set of claims obtained by replacing an extra structure symbol by expressions):

**Axioms of functions.** *For any term  $t$  with one argument, any values of its parameters, any set  $E$  where  $t$  is valid, and any functions  $f$  and  $g$ , the first of these axioms summarizes the next 3 ones:*

$$f = (E \ni x \mapsto t(x)) \Leftrightarrow (\text{Dom } f = E \wedge (\forall x \in E, f(x) = t(x)))$$

$$\text{Dom}(E \ni x \mapsto t(x)) = E$$

$$\forall x \in E, (E \ni y \mapsto t(y))(x) = t(x)$$

$$(\text{Dom } f = \text{Dom } g \wedge \forall x \in \text{Dom } f, f(x) = g(x)) \Rightarrow f = g$$

### 1.9. Set generation principle

Bounded quantifiers give sets their fundamental role of ranges of bound variables, unknown of the predicate  $\in$  which only sees them as classes. Technically,  $\in$  is definable by  $x \in E \Leftrightarrow (\exists y \in E, x = y)$  but cannot suffice in return to define bounded quantifiers by bounded formulas. Philosophically, the perception of a set as a class (classifying each object as belonging to it or not) does not provide its effective perception as a set (the perspective over all its elements as coexisting).

**Principle.** *For any defined class  $\mathcal{C}$ , if the expression  $(\forall x, \mathcal{C}(x) \Rightarrow \mathcal{R}(x))$  of  $\forall_{\mathcal{C}}$  on an undefined unary predicate  $\mathcal{R}$  is proven equivalent to a bounded formula here written  $(Qx, \mathcal{R}(x))$ , then  $\mathcal{C}$  is a set that we can name by a new operator symbol  $K$  of set theory, with arguments the common parameters of  $\mathcal{C}$  and  $Q$ , and axioms : (for all accepted values of parameters),  $(\forall x \in K, \mathcal{C}(x))$  and  $(Qx, x \in K)$ .*

The equivalence of  $Q$  and  $\forall_C$  is reducible to the following system where  $(\overline{Q}x, \mathcal{R}(x)) \not\equiv (Qx, \neg\mathcal{R}(x))$ :

- (i)  $\forall x, (\mathcal{C}(x) \Leftrightarrow \overline{Q}y, x = y)$  (in fact we just need  $\forall x, \mathcal{C}(x) \Rightarrow \overline{Q}y, x = y$ )
- (ii)  $Qx, \mathcal{C}(x)$
- (iii) For all unary predicates  $\mathcal{A}$  and  $\mathcal{B}$  such that  $\forall x, \mathcal{A}(x) \Rightarrow \mathcal{B}(x)$  we have  $(Qx, \mathcal{A}(x)) \Rightarrow (Qx, \mathcal{B}(x))$ .

Indeed these 3 properties are already known consequences of “ $Q = \forall_C$ ”. Conversely,

$((ii) \wedge (iii) \wedge (\forall_C x, \mathcal{R}(x)) \Rightarrow Qx, \mathcal{R}(x))$   
 $((i) \wedge \exists_C x, \mathcal{R}(x)) \Rightarrow \exists x, ((\overline{Q}y, x = y) \wedge (\forall y, x = y \Rightarrow \mathcal{R}(y))) \Rightarrow ((iii) \Rightarrow \overline{Q}x, \mathcal{R}(x)).$   $\square$

(iii) will often be immediate, by lack of any troubling occurrence of  $\mathcal{R}$  in  $Q$  (negation, equivalence, left of a  $\Rightarrow$ ), leaving to verify (i) and (ii). Counter-example:  $\overline{Q} = \mathcal{C} = 1, Q = 0$  does not satisfy (ii).

Here are examples of such operator symbols (denoting  $D = \text{Dom } f$ ):

$K$	$\{y \in E   \mathcal{B}(y)\}$	$\bigcup E$	$\text{Im } f$	$\emptyset$	$\{y\}$	$\{y, z\}$
$V_K$	$\forall x \in E, V_{\mathcal{B}}(x)$	$\text{Set}(E) \wedge \forall x \in E, \text{Set}(x)$	$\text{Fnc}(f)$			
$\mathcal{C}(x)$	$x \in E \wedge \mathcal{B}(x)$	$\exists y \in E, x \in y$	$\exists y \in D, f(y) = x$	0	$x = y$	$x = y \vee x = z$
$Qx, \mathcal{R}(x)$	$\forall x \in E, \mathcal{B}(x) \Rightarrow \mathcal{R}(x)$	$\forall y \in E, \forall x \in y, \mathcal{R}(x)$	$\forall x \in D, \mathcal{R}(f(x))$	1	$\mathcal{R}(y)$	$\mathcal{R}(y) \wedge \mathcal{R}(z)$
$\overline{Q}x, \mathcal{R}(x)$	$\exists x \in E, \mathcal{B}(x) \wedge \mathcal{R}(x)$	$\exists y \in E, \exists x \in y, \mathcal{R}(x)$	$\exists x \in D, \mathcal{R}(f(x))$	0	$\mathcal{R}(y)$	$\mathcal{R}(y) \vee \mathcal{R}(z)$

The definition of  $K = \{x \in E | \mathcal{B}(x)\}$ , that was only expressed as a class, translates as  $((\forall x \in K, x \in E \wedge \mathcal{B}(x)) \wedge (\forall x \in E, \mathcal{B}(x) \Rightarrow x \in K))$ , or as  $(K \subset E \wedge \forall x \in E, x \in K \Leftrightarrow \mathcal{B}(x))$ .

The functor  $\bigcup$  is the union symbol, and its axioms form the *axiom of union*.

For all function  $f$ , the set  $\text{Im } f$ , called the *image* of  $f$ , is the set of values of  $f(x)$  (range of the variable  $f(x)$ ) when  $x$  browses  $E$ . For all sets  $E$  and  $F$ , a *function from  $E$  to  $F$*  is a function  $f$  such that  $(\text{Dom } f = E)$  and  $(\text{Im } f \subset F)$ . An  $F$  satisfying this last condition (that can be written  $\forall x \in \text{Dom } f, f(x) \in F$ ) is called a *range of  $f$* .

The *empty set*  $\emptyset$  is the only set without element. For all set  $E$ ,  $(\emptyset \subset E)$ , thus  $(E = \emptyset \Leftrightarrow E \subset \emptyset \Leftrightarrow \forall x \in E, 0)$  and thus  $(E \neq \emptyset \Leftrightarrow \exists x \in E, 1)$ . This first symbol of constant ensures the existence of a set; conversely, it is reobtained by  $\emptyset = \{x \in E | 0\}$ .

We can reobtain  $\exists$  by  $(\exists x \in E, \mathcal{R}(x)) \Leftrightarrow \{x \in E | \mathcal{R}(x)\} \neq \emptyset \Leftrightarrow (1 \in \text{Im}(E \ni x \mapsto \mathcal{R}(x)))$ .

As  $(\text{Dom } f = \emptyset \Leftrightarrow \text{Im } f = \emptyset)$ , the only function with domain  $\emptyset$  is called the *empty function*.

For all  $x$ ,  $\{x, x\} = \{x\}$ . Such a set with only one element ( $x$ ) is called a *singleton*.

The set  $\{x, y\} = \{y, x\}$  is a *pair* (set with two elements) when  $x \neq y$ .

Our set theory will later be completed with more symbols and axioms, either necessary (as here) or optional (opening a diversity of possible theories).

## Metamathematical Complements

### 1.A. Completeness and Incompleteness Theorems

Let us sketch these two theorems by Gödel of key importance to the foundations of mathematics.

In a theory  $T$ , a *proof* of a formula  $A$  is a finite system, model of the *proof theory*, connecting  $A$  to (a finite number of) axioms of  $T$ , ensuring the truth of  $A$  in every model of  $T$ .

We say  $A$  is *provable* and write  $T \vdash A$  if a proof of  $A$  exists; *refutable*, if  $T \vdash \neg A$ ; *undecidable* if it is neither provable nor refutable. A theory  $T$  is said *contradictory* if it has a formula both provable and refutable, but then all are so ( $T \vdash 0$ ); otherwise it is *consistent*.

The completeness theorem gives model theory its strength, establishing the equivalence between realism and formalism for generic theories. Proven for a specific formalism, it makes the concept of provability perfect and independent of the choice of formalism with the same quality:

**Completeness Theorem.** *Any consistent generic theory has a model.*

*Proof.* Write each axiom as a chain of quantifiers applied to a quantifier-free formula. Replace each  $\exists$  there by a new operator symbol  $K$  by the rule  $(\forall x, \exists y, \forall z, A(x, y, z)) \rightarrow (\forall x, \forall z, A(x, K(x), z))$ . Add one by one to the axioms each closed formula (predicate symbol over terms) consistent with previous axioms. As all get Boolean values without contradiction, the set of closed terms forms a model.  $\square$

Any formula  $A$  not refutable in  $T$  is true in some model of  $T$  (a model of the consistent theory  $T + A$ ). So,  $(T \vdash A) \Leftrightarrow (A \text{ true in all models of } T)$ . As models built this way depend on an arbitrary order between formulas, an infinity of variants is possible. But no founding theory (such as a set theory) can specify all its own truths (ideal values of its closed formulas):

**Truth Undefinability Theorem.** *Let  $T$  a theory defining in each model  $M$  the invariant data of :*  
– “a theory”  $T'$  where each expression  $F$  of  $T$  meta-invariantly translates as  $[F]$  in  $T'$ ,  
– a unary predicate  $v$  to apply to the image  $[F]$  of each closed formula  $F$  of  $T$ ,  
Then there exists a closed formula  $G$  of  $T$ , such that  $T \vdash (G \not\leftrightarrow v[G])$ .

Proof. Assume the image  $[F]$  of each formula  $F$  of  $T$  is named by a closed term  $t_F$  of  $T$ , whose construction can be formalized by a unary term  $J$  of  $T$ :  $J([F]) = [t_F]$ . In  $T'$ , replacing the variable of a formula  $A$  by a closed term  $u$  forms a closed formula  $(A : u)$ . So,  $([F] : [K]) = [F(K)]$ .

Let  $H(A)$  be the formula  $\neg v(A : J(A))$ , with variable  $A$  in the class of unary formulas of  $T'$ . Let  $G$  be the formula  $H(t_H)$ . Then  $[G] = ([H] : J([H]))$ , thus the conclusion.  $\square$

**Corollary.** *If  $M$  is explicitly built (meta-invariant) determining values of closed formulas  $F$  of  $T$ , then their computation  $v[F]$  in  $M$  by the same rules (if possible) is incorrect.*

Proof.  $G$  and  $v[G]$  are computed by the same rules but respectively interpreted outside and inside  $M$ . As  $G \not\leftrightarrow v[G]$ , if the outside computation is correct, then the one in  $M$  (thus some concept involved; that is provability we used for the completeness theorem) is not.  $\square$

**Incompleteness Theorem.** *If  $T'$  is built like  $T$ , and  $T$  can express the provability  $p$  in  $T'$ , then:  $(T \vdash F) \Rightarrow (T \vdash p[F])$ , and  $(T \vdash \neg p[0]) \Leftrightarrow (T \vdash 0)$ .*

The proof of  $F$  in  $T$  can be copied into  $T'$ , or converted to a proof of existence of a proof.

Let  $T \vdash (G \not\leftrightarrow p[G])$ . Then  $(T \vdash G) \Leftrightarrow (T \vdash (G \wedge p[G])) \Leftrightarrow (T \vdash 0)$ . Thus  $T \vdash (p[G] \Leftrightarrow p[0])$ . Thus  $(T \vdash \neg p[0]) \Leftrightarrow (T \vdash G) \Leftrightarrow (T \vdash 0)$ .  $\square$

Contrapositive viewpoint if  $T$  can describe models: every model of some  $T'$  is a model of  $T$ .

No converse: a proof in  $T'$ , model of proof theory that is “finite” according to  $T$ , may actually be infinite and thus no real proof. If a formula  $F$  only has infinite “proofs”, its provability is undecidable and false: the unfound proofs of  $F$  seem to not have been sought long enough, and models where  $F$  is false cannot be reliably built, so “do not always exist”. Wherever models of  $T$  can be found, some of them cannot find any one of them anymore.

## 1.B. Metamathematical Time and Zeno’s Paradox

The mathematical world’s way to think of itself, is similar to ours.

Describing the world and the language that describes it, means describing everything, and something beyond it. Mentioning “what I mean”, does not itself say what it is, as it might be anything, and becomes absurd if the phrase modifies or contradicts it (“the opposite of what I’m saying”).

Mentioning “what I will tell about tomorrow”, even if I knew what I will say, would not provide its meaning either: if ever I will mention “what I told about yesterday” (thus now) it would make a vicious circle; but even if the structure of my future words ensured that its meaning will exist tomorrow, it would still not provide it today. However I may speculate on it, its actual meaning will only arise once actually expressed in context. The lack of interest to discuss words without their meaning, makes it better to ignore the present and future sayings, and focus on the past ones.

Like historians, I can only describe the universe of the past, being myself in a present outside this universe. I can speak of “what I told about at that time”: it has a sense if those past words already had one, because I got that meaning and I remember it.

My current universe of the past that I can describe today, includes the one of yesterday, but also my yesterday’s comments about it and their meaning. So I can describe today things outside the universe I could describe yesterday. While I neither learned to speak Martian nor acquired a new transcendental intelligence since yesterday, the same language applies to a broader universe with new objects. As these new objects resemble the old ones, my universe of today can resemble my universe of yesterday; but from one universe to another, the same words can take a different meaning.

Achilles runs after a turtle; every time he crosses the distance to it, it takes a new length ahead. Seen from a height, a vehicle gone on a horizontal road approaches the horizon. Particles are sent in accelerators closer and closer to the speed of light. Can they reach their ends ?

Each example can be read two ways: the one, “closed”, sees a reachable end; the other, “open”, excludes it to only see the movement approaching it indefinitely. Finally, a physical measurement dictates to each case its sole “true” reading (not the same). But the world of mathematics, where objects only have conventional roles, can accept both interpretations.

Each generic theory is “closed”, as it can see its world (range of its variables) as a set, and can prove any consequence of its axioms. But itself (or any theory able of founding it) escapes this whole.

Hence the need for an open theory integrating each theory of a past world as an object of a later world, forming an endless series of growing realities. This open theory will be set theory.

### 1.C. The relative sense of open quantifiers

As an exclusion of ends, set theory excludes open quantifiers from its formulas: once the free variables and the content of involved sets are fixed, its bounded expressions (expressing local, explicitly subjective questions) make sense regardless of rest of the universe (which may keep growing, only constrained by the chosen axioms).

In granting no sense to some claims, it acknowledges their relative, contingent nature, only valid “here and now”. A universal claim  $(\forall x, \dots)$  true “here”, may become false (a counterexample might arise) “elsewhere”. But this dependence can not be affirmed: the question of how are things “elsewhere” would depend on the variation range of the universe, and is reducible to a feature of their union:  $(\forall x, \dots)$  is true “in all universes” if it is true in the union of all universes. Indeterminacies should only be considered by avoidance, as equally unknown as the panorama of the accepted universes they refer to, partially selected by axioms.

The completeness theorem identifies this indeterminacy to the limits of formal provability. A claim is provable when its negation entails a contradiction, where the negation exchanges  $\forall$  and  $\exists$ , and each is translated by the rules of proof we mentioned. In any class,  $((\exists x, \mathcal{A}(x)) \wedge (\forall x, \mathcal{B}(x))) \Rightarrow \exists x, \mathcal{A}(x) \wedge \mathcal{B}(x)$ , so that  $(\exists x, \mathcal{A}(x))$  and  $(\forall x, \neg \mathcal{A}(x))$  can not be both provable without contradiction. A proof of  $\exists x, P(x)$  is made of a term  $t$  and a proof of  $P(t)$ , or terms  $t, t', t'' \dots$  and a proof of  $(P(t) \vee P(t') \vee P(t'') \dots)$ .

The undecidability of  $\exists x, P(x)$  can come from universes  $\mathcal{U} \subset \mathcal{U}'$  where the claim is only true in  $\mathcal{U}'$ , and the  $x$  satisfying  $P(x)$  are all outside  $\mathcal{U}$ . Intuitively, these objects are out of reach by determined means (terms) or indetermined ones (chosen existence axioms).

But there are other possibilities: some universes where it is false may not be extended to ones where it is true (satisfying axioms), and a universe where it is true may not include a class of objects  $x$  all satisfying  $\neg P(x)$ , which preserves the false value of  $P(x)$  and satisfies all axioms (which, with operators, may recreate some  $x$  satisfying  $P(x)$ ). Thus, the different possible universes with different properties do not necessarily follow one another in time, but may belong to separate, incompatible dynamics, some more realistic than others.

### 1.D. Nature of classes and the set generation principle

The equality between classes  $\mathcal{A}$  and  $\mathcal{B}$  would be defined by  $\forall x, \mathcal{A}(x) \Leftrightarrow \mathcal{B}(x)$ . In particular, any universal claim  $(\forall x, \mathcal{A}(x))$  means the equality of  $\mathcal{A}$  to the universe. This notion of equality is as indeterminate as the open  $\forall$ , thus must be abandoned. But the notion of *proven equality* between classes remains, given by a proof of equivalence between formulas. Classes do not behave like objects.

Each universe  $\mathcal{U}$  sees each class  $\mathcal{C}$  as a meta-set of objects  $P = \{x \in \mathcal{U} | \mathcal{C}(x)\}$ ; and calls it a set when  $P \in \mathcal{U}$  (it has the same elements as an object “set” in  $\mathcal{U}$ ). Otherwise, this  $P$  is being born (emerging) now and will exist as a set from now on in the next universes.

From the perspective of a variable universe, a class  $\mathcal{C}$  is called a set (equal to  $P$ ) if this  $P$  turns out to not depend on  $\mathcal{U}$ , thus when no “unknown” object  $x$ , outside the current universe (the “first”  $\mathcal{U}$ ) can ever satisfy  $\mathcal{C}(x)$ . Instead, a class beyond sets remains able to contain elements that do not yet exist: the formula  $\mathcal{C}$  lets a chance for foreign elements to satisfy it, and thus to join  $P$  and make it vary during the growth of  $\mathcal{U}$ .

Finally, once the variation of  $\mathcal{U}$  receives a fixed range, the open formula qualifying a class  $\mathcal{C}$  as a set,  $(\exists E, \forall x, \mathcal{C}(x) \Rightarrow x \in E)$ , interpreted in the “ultimate” universe made of the union of these  $\mathcal{U}$ , expresses that in this variation, “there is a time after which  $P$  is constant”, thus with the advantage of not trusting the first universe. But as the variable  $\mathcal{U}$  escapes all sets, the different perspectives all exceed one another, and the same class may alternatively gain and lose the status of set.

#### *Justification of the set generation principle*

Let  $\overline{Q}$  a quantifying formula such that  $\neg(\overline{Q}y, 0)$ , and let  $\mathcal{C}$  the class of  $x$  such that  $\overline{Q}y, y = x$ . For all  $x$  in  $\mathcal{C}$  we have  $(\overline{Q}y, y = x)$  but  $\neg(\overline{Q}y, 0)$ . Now  $\overline{Q}$  has only finite, fixed means independent of  $x$  to distinguish the predicate  $(y \mapsto (y = x))$  from  $(y \mapsto 0)$ : a finite formula, variables bound to sets, fixed parameters, providing the values of  $y$ , where to test the predicate. This can no more distinguish both predicates as soon as  $x$  is an alien (not tested by  $\overline{Q}$ ). Thus  $x$  is an Earthling. As all its elements are Earthlings,  $\mathcal{C}$  is a set.  $\square$