The FMathL mathematical framework

Arnold Neumaier

Fakultät für Mathematik, Universität Wien
Nordbergstr. 15 , A-1090 Wien, Austria
email: Arnold.Neumaier@univie.ac.at
WWW: `http://www.mat.univie.ac.at/~neum`

Version 1.04, October 30, 2009

Please quote the version and date when referring to this version of the document. The newest version will always be available at
`http://www.mat.univie.ac.at/~neum/ms/fmathl.pdf`

This is a working copy that will be revised from time to time as feedback is incorporated, in order to make the FMathL mathematical framework optimally accepted in the community of mathematicians, best adapted to the actual practice of mathematics, easy to comprehend, and easy to use.

Please pass this on to colleagues who are likely to take an interest and may give feedback from different points of view.

Abstract. Several frameworks for mathematics have been constructed in the literature. To avoid the paradoxes of naive set theory, Zermelo and Fraenkel constructed a type-free system of sets, based on first order predicate logic, while von Neumann, Bernays and Gödel constructed a system with two types, sets and classes. These and related systems suffice for the common needs of a working mathematician. But they are inconvenient to use without pervasive abuse of language and notation, which makes human-machine communication of mathematics difficult.

In this document, we construct a new framework for mathematics, designed as part of the specification system FMathL for the formalized communication of mathematics between humans and machines, in a way close to the actual practice of mathematics. The framework is described in such a way that this description can become part of the FMathL system itself.

All axioms are given in the form of familiar existence requirements and properties that are used by mathematicians on an everyday basis. Thus mathematicians trusting the axiom of choice and hence the law of excluded middle can readily convince themselves of their truth in their private (but publicly educated) view of mathematical concepts.

The exposition is such that students exposed to the typical introductory mathematics courses should have no serious difficulty understanding the material.

Contents

Chapter 1

Motivation and informal set-up

1.1 Introduction

The FMathL mathematical framework is designed to be a formal framework for mathematics that will allow – when fully implemented in some programming language – the convenient use of and communication of arbitrary mathematics (including logic) on a computer, in a way close to the actual practice of mathematics, with emphasis on matching this practice closely. The acronym FMathL is an abbreviation for **Formal Mathematical Language**. General background material for the FMathL approach to mathematical modeling can be found on the web site [54].

FMathL looks like the mathematics accessible to undergraduates, and it specifies only concepts and properties that are (or should be) familiar to any mathematics student who mastered the standard introductory courses. Of course, only those concepts are discussed that are basic in the sense that they form the raw material for mathematical discourse, while everything else (e.g., groups and vector spaces) can be constructed on their basis in a straightforward way.

FMathL is an axiomatic framework for mathematics that formalizes the essential properties of the basic mathematical objects and only these. The user of FMathL as a foundation of mathematics will know by the design of FMathL which properties are essential in that they can be relied upon in each implementation, and which properties cannot be relied upon since, being implementation-dependent, they are accidental byproducts of a particular implementation.

The present document constitutes the syntax-free, abstract part of the FMathL mathematical framework, defining the **object level** of mathematics by giving axioms specifying the properties required of mathematical **objects**. Included is that basic part of mathematics that is needed to be able to formally write in an easily readable form everything books on predicate logic, axiomatic set theory, and calculus (which we may regard as constituting traditional foundations) need to be able to define their subject. This is slightly more than the minimum that needs to be available to discuss the foundations in terms of itself, a process I call reflection.

There will also be a specification part, with a concrete syntax given in terms of an explicit grammar (Neumaier et al. [58]), and an implementation part, which describes how to implement the concepts used to define the object level by reflection inside the object level; cf. the companion paper by Neumaier & Schodl [57]. Ultimately, there will be also an implementation in some programming language, so that any (human or computer) system that understands this programming language will be able to process FMathL and will thus understand mathematics, in the sense that it can perform like a mathematician all routine aspects of mathematics.

The separation into these parts has several advantages: It frees us from the need to formally discuss syntactical issues (grammars and parsing) within the abstract part. It frees us from the need to consider implementation issues within the specification part. It allows different modes of specification and implementation of concepts in FMathL.

Chapter 2 and Section 3.7 contain the mathematically relevant part of the document. We only prove a few theorems. But we mention without proof many familiar properties that can be deduced from the axioms given in a straightforward way. The remaining sections mostly discuss the motivation for and the philosophy of FMathL and its design.

In the remainder of Chapter 1, we discuss some aspects related to the design of the FMathL mathematical framework. This discussion is informal, and touches – as any foundational system must – a number of philosophical questions regarding mathematics and how it is practiced. In particular, we clearly differentiate between the subjective and objective aspects of mathematics.

Chapter 2 then builds the axioms and definitions that constitute the abstract version of the framework. To complete the foundations of mathematics, Chapter 3 rounds the picture by discussing some further basic terminology and their implications, without introducing further axioms.

Acknowledgments. I want to thank the members of our FMathL seminar for many discussions on the subject, in particular Hermann Schichl and Flaviu Marginean for playing the role of advocatus diaboli during the evolution of the present material, and Peter Schodl, Kevin Kofler, Waltraud Huyer, and Mihaly Markot for commenting on earlier versions of the manuscript. I want to thank my brother Wilfried Neumaier for discussions about Ockham's razor and nominal objects, and their history, Martin Bunder for discussions on illative combinatory logic, and Andries Brouwer, Martin Fuchs, Johannes Schoissengeier, and Mike Shulman for useful comments on earlier versions of the document.

Partial support by the Austrian science foundation FWF under contract number P20631 is gratefully acknowledged.

1.2 The need for a mathematical framework

The December 2008 *Special Issue on Formal Proof* of the Notices of the American Mathematical Society [1] features a collection of articles by Thomas Hales, Georges Gonthier, John Harrison, and Freek Wiedijk, practitioners and theorists of formal proofs and computer-assisted proofs. These articles explore the use of computers in proofs, both extending mathematics with new results and creating new mathematical questions about the nature and technique of such proofs.

The time seems ripe for creating the foundations for an automatic mathematical assistant that combines the reliability and speed of a computer with the ability to perform at the level of a good mathematics student, with respect to mathematical knowledge and proofs, the ability to understand ordinary mathematical language and to solve standard exercises, the ability to organize mathematical information, and a searchable database of mathematical knowledge, so that it provides advanced support for students, professional researchers, and industry in mathematical work.

Mathematicians know that they already have a very expressive language optimized for expressing things in any desired degree of brevity or detail, for maximal intelligibility and easy overview. The common mathematical language must be learnt by everyone anyway, hence it should be the basic language for any intelligent system designed for the ordinary scientist.

To gain quick acceptance among mathematicians, a mathematical research system implementing such an assistant must therefore be based on mathematics as mathematicians practice it, with their language, their editing tradition, and their informality. The system must be both simple on the low level, building semantics into a very low-level representation, and friendly on the high level, building high-quality interfaces.

Barendregt [6] and Beeson [10] are recent surveys of the history and different styles of current proof assistants (automatic theorem provers); none of these proof assistants has the user-friendliness required to make these systems routinely useable by mathematicians. Current semantic computer support for ordinary mathematics notation in projects such as MathML (Miner et al. [17]) is improving but still quite limited; see Kofler et al. [39] for a number of current limitations.

Within our FMathL project, the computational mathematics group of the University of Vienna is preparing the ground for a mathematical research system called MathResS (pronounced "mattress", for a good, comfortable foundation of mathematics on the computer). More about the FMathL project, its vision, and partial results towards this goal can be found at the project web site `http://www.mat.univie.ac.at/~neum/FMathL.html#FMathL`

MathResS is designed explicitly with the goal that scientists will like to use it because it provides mathematical contents and proof services on their laptops

– as easily as Latex provides typesetting service, the arXiv provides access to preprints, Google provides web services, Matlab provides numerical services, or Mathematica provides symbolic services –, in a way that they can easily comprehend and that, unlike with present systems for formalized mathematics, does not take undue amounts of extra time on their part.

As a first step, one needs to create a conceptual basis for mathematical notions that is close to the actual practice of mathematics but specified unambiguously and precisely enough so that it is programmable to make it understandable by a machine without an excessive, human-assisted overhead in presenting mathematics of interest to the system.

Formalized mathematics. Doing mathematics directly and fully formalized in ZFC (Zermelo-Fraenke set theory with the axiom of choice), say, is like doing application programming on a Turing machine: Already the simplest standard practices become long, messy, nearly incomprehensible, error prone, and very time consuming.

According to Barendregt & Wiedijk [8], creating a complex mathematical document from the decision to write it (but after the results were already more or less obtained) to final publication currently takes about 4 hours/page. (In my experience, this includes the time needed for material selection, writing, correction, proofreading, reading and answering referee's reports, etc.) Creating a corresponding formally verified document with current systems takes, according to the same source, about 10 times as long, and the result is nearly indigestible for a human.

In my opinion, this huge overhead – which makes these systems unattractive for all but the most determined potential users – is due to the huge distance between the input the current generation of human-assisted automatic theorem provers can accept and the input a human theorem prover accepts. One reason for this is that current theorem provers express mathematics in terms of very concept-parsimonious foundational systems, often some form of axiomatic set theory.

In contrast, ordinary mathematical language has a rich collection of concepts adapted to optimally expressing what mathematicians want to say. Anyone who has tried to reduce some nontrivial but elementary mathematics – such as the construction of the field of rational numbers – rigorously to set theory knows that this reduction is both error-prone and results in very lengthy and hardly digestible formulas, looking much more like low level programming code than like ordinary mathematics.

This is why, in practice, mathematicians pay only lip service to current foundations. Not even Nicolas Bourbaki's foundational book *Theory of sets* (Bourbaki [13]), the pinnacle of mathematical rigor, proceeded far with fully formalized set theory, and uses a host of – from the point of view of formalized mathematics based on ZFC forbidden – **abuses of language** and **abuses of notation**, justifying it in the introduction with words such as *"But formalized mathematics cannot in practice be written down in full [...] We shall therefore very quickly* *abandon formalized mathematics"*, or *"As far as possible we have drawn attention to abuses of language, without which any mathematical text runs the risk of pedantry, not to say unreadability"*, while still claiming that *"our series lays claim to perfect rigor: a claim which is not in the least contradicted by the preceding considerations, nor by the need to correct errors which slip into the text from time to time."*

Since mathematics as an easily communicated discipline depends a lot on such abuses of language, set theory, while sufficient to provide a foundation of mathematics "in principle", is unable to fill the role of a foundation of mathematics efficiently enough for its easy use on a computer.

Nothing much has changed since Bourbaki's treatise. In a recent (2007) review of the QED manifesto [62], a well-known 1994 vision of the theorem proving community, Freek WIEDIJK [71] remarks: *"Currently formalized mathematics does not resemble real mathematics at all. Formal proofs look like computer program source code. For people who do like reading program source code that is nice, but most mathematicians, the target audience of the QED manifesto, do not fall in that class."* – *"If one writes a formalization of a mathematical result, then one has to work quite hard, and then at the end one has a* `tar.gz` *file with several computer program-like files in it. However, unlike a computer program, those files have no immediate further use. The fact that they fully describe the mathematics has some aesthetic appeal, and it is nice that they make it completely certain that the mathematics is correct, but the unformalized version of the result already was beautiful and understandable, so not much is gained."* – *"If we want to make some progress of getting people actually to use formal mathematics, it has to be close to the way mathematics already is being done for centuries."*

This is why, with FMATHL, I opted for designing a new mathematical framework that gives up the principle of maximal parsimony of concepts in favor of a more generous basis that makes life easier for the user.

Among other things, this requires that the FMATHL mathematical framework must be aware of all possible ambiguities in the common mathematical language, and have ways to either avoid these ambiguities or to resolve them from the context. Thus, in this document, we pay attention to language constructs that lead to ambiguity.

1.3 Some design principles

In this section, we discuss some important principles forming the background for the design of the FMATHL mathematical framework. Initially, the main concern was just to get a concise, flexible, and easy to use formal representation for mathematics on a computer, but it turned out that to achieve this, a full representation of complete foundations of mathematics was needed.

Traditional foundations. To clarify the distinguishing features of the FMathL approach, we contrast it with some traditional foundational systems, namely **ZFC**, **Zermelo–Fraenkel set theory** with the axiom of choice (Zermelo [74], Fraenkel [26], Skolem [68]), **illative combinatory logic** (Schönfinkel [66], Curry & Feys [20]), and, to a smaller extent, **CCAF**, the **category of categories as a foundation** (Lawvere [46, 47]), which all provide ontologies for the concepts of mathematics.

ZFC, a system of first order predicate logic for sets with a single nonlogical relation $\in$ (membership) builds mathematics upon the primitive notion of a set. Every mathematical object (including numbers, pairs, and functions) is regarded as a set. Infinitely many axioms, expressed by a finite axiom scheme generating these axioms, specify the properties of the membership relation. ZFC is regarded by the majority of today's mathematicians (if pressed for a commitment) as *the* accepted foundation of mathematics.

Combinatory logic is an algebra with a single operation (application), three distinguished objects, the combinators I, K, and S, and three simple axioms governing the behavior of the combinators. It serves as a model for the **λ-calculus**, whose illative extensions (see, e.g., Bunder [15]) build mathematics upon the primitive notion of a function. Every mathematical object (including numbers, pairs, and sets) is regarded as a function. The illative version is a bit less prescriptive in this respect, calling general objects obs, but functions and classes (both are special obs) overlap in a nonstandard way.

In the (differing) versions of the categorical foundations CCAF given by Makkai [50] and by McLarty [52], there is a distinction between numbers, simple sets, and functions. (In the original version [46] of the set part of McLarty's version of CCAF, called **ETCS**, the **elementary theory of the category of sets**, every set is even regarded as a mapping.) But since, in ETCS, a set cannot contain another set, what mathematicians regard as sets of sets are no longer sets but functions!

Clearly, these ontologies cannot be true simultaneously. In ZFC, not every set is a function, while in the λ-calculus (as in the set theory of von Neumann [59], the precursor of **NBG**, the **von Neumann–Bernays–Gödel theory** of classes), not every function is a set, and in the categorical approach, things depend a lot on which author one follows.

Thus these ontologies make contradictory assertions about the interpretation of objects. The ontologies are equivalent only in a very weak sense. While they all allow to represent arbitrary mathematical concepts with their characteristic properties, the objects having these properties have additional accidental properties extraneous to mathematical agreement, and the objects may differ in these accidental properties when represented in the different ontologies. (See Section 2.7 for further accidental properties of different constructions of the concept of a real number within ZFC.) And to map them to the current categorical foundations, one even needs to translate sets to different kinds of objects, depending on which sort of elements they contain.

Interpreting Ockham's razor. This is an unsatisfactory state of affairs for proper foundations of mathematics. Good mathematical foundations should have a precise specification that tells which properties of a mathematical object are essential, i.e., characteristic and indispensable for their use. They should not allow one to make any inferences about accidental properties that are extraneous to mathematical usage. They should not need a translation process that leads from the concepts as used in practice to the concepts as demanded by the foundations.

The situation is analogous to the definition of a list in computer science. One might give a definition in terms of the way a list is represented and accessed in the memory of a computer; this may differ in different implementations. However, what counts for the user are only a few characteristic properties of a list, namely those that allow it to perform its intended role. These properties (defining the interface) can be specified without knowing the implementation (how they are realized in a particular environment), and tell the essence of what a list is. On the other hand, it is good to know about different implementations since this reveals something about implementability of lists, which may matter for efficiency.

Similarly, ZFC, combinatory logic, and category theory provide nonisomorphic models implementing – very parsimoniously, but from a computational point of view very inefficiently – some common structure that captures the essence of mathematical concepts. The essence is precisely the part that must be represented in order for the model to deserve being called an implementation of mathematics. (See AWODEY [5] for a discussion of the essence, there called mathematical content, from a categorical point of view.)

One goal of the present document is to present an axiomatic framework for this common structure that formalizes the essential properties and only these. FMATHL makes available in a precisely defined way the full expressivity of standard mathematics in the form taught at universities and used by mathematicians, but without any accidental requirements. (See MCLARTY [51] for an alternative, categorical approach with similar features as regards accidental features, but the approach is described in terms that are unfamiliar to many mathematicians.)

The minimalistic approach used in ZFC and combinatory logic, where everything is reduced to a minimal number of initial concepts from which all others are constructed, is an embodiment of a particular use of a basic principle of modern science, *"frustra fit per plura quod potest fieri per pauciora"*, quoted in Newton's Principia Mathematica, that we should not use more degrees of freedom than necessary to model a phenomenon. Today, it is called **Ockham's razor**, after William of OCKHAM [60], the most prominent medieval logician, although the principle goes back to Ockham's teacher Duns Scotus; cf. BUCKNER [14].

Noticing that foundations must be primarily clear, unambiguous, and objective, FMATHL uses Ockham's razor in a different way: Instead of cutting away as many concepts as possible, FMATHL aims at cutting away all accidental, subjective properties of concepts that are introduced by the minimalistic approaches.

A specification approach. The FMATHL mathematical framework therefore takes a specification-oriented axiomatic approach in which the basic mathematical concepts are specified by their properties rather than by a construction from primitives where the objects inherit unwanted accidental properties.

As the concept of a vector space allows a basis-independent development of linear algebra by abstracting from specific coordinates, although the latter may be introduced for specific constructions, so the FMathL approach abstracts from the specific implementation of mathematical concepts, although the concrete implementation is important for each particular subject.

Systems like ZFC, combinatory logic, or CCAF may then be seen as possible subject levels for the implementation of a mathematical framework, rather than as their ontology.

As a result, in contrast to ZFC, combinatory logic, and CCAF, the FMathL mathematical framework does not have minimal symbolic foundations without any redundancy; instead, the foundations are generous, easy to comprehend, and reflect easy specification principles on the highest possible level.

Every elementary activity that a mathematician may want to perform is reflected in our axiomatic system. For example, mathematicians often pick an element x from a set A. In FMathL, the resulting element is denoted by Choice(A) (Bourbaki [13] writes εA in the Hilbert tradition), and its obvious properties are specified by Axiom A19.

Reasoning in mathematics is – from a computational point of view – comparable to reasoning about a complex programming language: for comfortable use, one needs to axiomatically require all constructs and their properties rather than reduce the latter to primitive operations on a Turing machine. The fact that there are many elementary activities that mathematicians perform naturally leads to correspondingly many constructors, and hence to an axiomatic system with many axioms.

All properties required axiomatically in FMathL are given in the form of familiar existence requirements and properties that are used by mathematicians on an everyday basis. Thus mathematicians trusting the axiom of choice and hence – cf. Section 3.1 – the law of excluded middle can readily convince themselves of their truth in their private (but publicly educated) view of mathematical concepts. This is the best approximation to the ancient ideal that axioms should be self-evident that I was able to achieve.

Note that all mathematicians already have (or should have) these concepts – or a close approximation of them – in their private implementation of mathematics. In this document, we refer to the private implementation of mathematics in a **subject** capable of understanding mathematics as the **subject level**. The subject level, the domain of brouwer's radical intuitionism, is clearly dependent on who is doing mathematics, But for our mathematical culture, the essential arena is where mathematics is communicated: Different subject levels have a common, though somewhat ambiguous informal language with which the subjects exchange their mathematical ideas, theorems, and proofs.

Foundational studies usually idealize the situation by abstracting from the differences in subject levels, and call the resulting idealized subject level the **metalevel** or the **background theory**, and its language the **metalanguage**. However, it is clear from the literature that different authors (and even different publications by the same author) use different metalanguages. Talking of subject levels recognizes this explicitly. Our exposition will make the point clear that mathematics is independent of the details of how it is implemented in different subjects, since equivalent mathematical frameworks can be defined in all subject levels in which common mathematical language is understood.

FMathL only codifies what is done anyway in mathematical practice. The exposition is such that students who were exposed to the typical introductory mathematics courses should have no serious difficulty understanding the material. They may look at an arbitrary page and will find that (with only few exceptions) they recognize all formulas as familiar mathematics or as simple definitions or exercises based on it. This makes it easy to trust the system.

Like in combinatory logic but unlike in ZFC or CCAF, every object whose existence is asserted can be constructed by means of a finite formula. This makes FMathL a suitable basis for the computer representation of mathematics. However, in contrast to combinatory logic, these formulas are much more transparent, representing their meaning more expressively, and they are very close to mathematical practice.

As a result, doing mathematics in the FMathL mathematical framework is like programming in a very user-friendly structured programming language – it even tolerates a certain amount of sloppiness without impairing understanding.

Consistency questions. Gödel's second incompleteness theorem [31] implies that no sufficiently rich consistent system of first order logic can prove its own consistency – though inconsistency is provable by exhibiting a contradiction. Therefore, consistency of a foundational system must be accepted on faith, fed by imagination and prior experience.

It may be argued that a minimalist approach makes it less likely that an undetected inconsistency is present. But the minimalistic naive set theory was for 25 years believed to be consistent before contradictions were discovered, such as Russell's paradox (Russell [65]) concerning the set of all sets not containing themselves.

This shows that inconsistency is difficult to predict even for simple systems. in particular, a necessary condition for trusting the consistency is that the axiom system is easy to comprehend in its consequences. For example, some axioms of ZFC are almost incomprehensible before one has introduced extra concepts and notation to build intuition. Moreover, the ZFC axioms must be carefully formulated, introducing nonintuitive restrictions that forbid the derivation of contradictions.

The situation is similar for combinatory logic, where the axioms are deceptively simple and intuitive. But – as history has shown – figuring out a system that can represent all mathematics without leading to inconsistency proved difficult and fallacious. Indeed, the origins of both the λ-calculus and combinatory logic were systems that – like naive set theory – at first looked fine but later turned out to be inconsistent [42, 19, 67].

The surviving canonical repaired versions of combinatory logic and the λ-calculus are restricted to describing recursive functions. In contrast to the situation in naive set theory, where ZFC, the standard repaired version, has the full mathematical expressivity, fully expressive consistent extensions of combinatory logic to illative combinatory logic (BARENDREGT et al. [7]) and other type-free languages (FEFERMAN [22, 23, 24]) do not seem to have reached a definite state where people agree on a standard reference version.

Thus it is not minimality or intuition, but time-honored familiarity with the consequences of an axiom system that makes us confident that the system is consistent, and hence useful for founding mathematical practice. But familiarity with the consequences requires comprehensibility, i.e., the expression of the axioms and their consequences in familiar terms.

FMATHL directly builds upon these familiar terms, thus represents a shortcut to comprehensibility, and hence to confidence in its consistency. The formulas of the FMATHL system are essentially those traditionally used by mathematicians, resulting in easy readability. Also, in contrast to ZFC, its axiom system is finite, though not as small as that of combinatory logic.

It may be possible to give relative consistency proofs of FMATHL relative to ZFC and/or illative combinatory logic by implementing a model of FMATHL in these systems. It may also be interesting to implement a model of FMATHL in FMATHL itself that uses only a minimal number of concepts and axioms from FMATHL. However, all this is outside the scope of the present document.

The best of all worlds. Like ZFC, FMATHL has a primitive concept of sets, a choice function, and a number of axioms directly inspired from ZFC, though expressed in a more readily digestible form.

Like illative combinatory logic, the FMATHL system builds mathematics on top of a propositional calculus emerging from the system itself, rather than on top of some predicate logic. It also takes from combinatory logic the need of functions as primitive objects and the idea of combinators; cf. Section 2.11.

Like CCAF, the FMATHL mathematical framework can be regarded as a category of all categories, though with additonal structure.

But the FMATHL mathematical framework restricts the extent to which different mathematical concepts are related to what is really necessary. In particular, unlike in traditional foundations, FMATHL does not make the assumption that a number or a function is a set, or that a set is a function. Thus – similar to a system of BEESON [9] – being a set or being a function are independent properties of mathematical objects in FMATHL, and there are important classes of primitive objects that are (without further assumptions) neither sets nor functions: statements, numbers and texts. On the other hand, FMathL considers pairs and tuples to be functions with the first few natural numbers as arguments, whose images are the first, second, etc. entry of the pair or tuple.

Assuming the consistency of ZFC, one can still discuss models of ZFC inside FMathL by working inside a ZF-algebra, a mathematical structure defined by the Zermelo-Fraenkel axioms (see, e.g., Awodey [4]). Every ZF-algebra contains natural ZF-numbers according to the usual construction, but these ZF-numbers are different for different ZF-algebras, and have nothing to do with the intrinsic natural numbers beyond the fact that both sets are countable. ZF-numbers are just one countably infinite set with a canonical well-ordering like many others, on each of which one can build a Peano system. (Indeed, the Peano axioms do not really define the natural numbers, but rather the concept of a counted infinite set!)

Concept ambiguity. Gödel's first incompleteness theorem (Gödel [31]) says that any sufficiently rich consistent system of first order logic contains a statement S that is true for an intended model of the system but cannot be proved in the system. Since the system obtained by adding the negated statement $\neg S$ to the original system is consistent (a proof of inconsistency would give a proof of S), it has a model by Gödel's completeness theorem (Gödel [30]), which says that everything valid in *arbitrary* models of a system of first order logic is provable. Therefore S is **independent** of the system: its truth value is undetermined, both in the sense that neither its truth nor its falsity can be proved, and in the sense that there are two models in which the truth values of S are different – and hence implementation-dependent, subjective.

This observation implies that any system based on classical first order logic in which one can formalize all mathematics will have inherent ambiguities, no matter how it is formalized. For a mathematical framework, the different models are like different possible subject levels – indistinguishable with respect to the specified properties of the object level, but possibly differing in things not decidable on the basis of these specifications.

Thus we may view undecidability as the lack of a complete specification of a concept: Only the decidable part of mathematics is clearly communicable and hence has an unambiguous, objective meaning. The remainder is implementation-dependent and inherently subjective since different subjects may give different answers, true in their private implementation.

The insight that a *completely unambiguous* specification of the precise contents of *all* mathematical concepts – by specifying a mathematical framework uniquely up to isomorphism – is impossible (at least in classical first order logic) frees us from the curse of having to strive for making everything unambiguous on the basis of the axioms given.

On the contrary, for the sake of convenient specifications, FMathL makes quite liberal use of this freedom, and decides as little as possible without losing the essence and usefulness of the common mathematical concepts. As usual in mathematics, more specific versions can be obtained by making additional assumptions beyond those specified in the common mathematical framework, such as the continuum hypothesis, or large cardinals, in an analogous way as number theorists assume the (so far undecided) Riemann hypothesis.

The lack of complete determination is common in the FMathL approach to the foundations, since FMathL concentrates on paving the roads travelled by mathematicians (i.e., on enabling them to easily specify the properties they actually use) rather than fencing them in by enforcing unnecessarily rigid rules (that try to eliminate all possible misuses of the formalism). In particular, we commit ourselves as little as possible to aspects deemed irrelevant for the actual usage of a concept.

Thus FMathL only formalizes the *common* ground of most mathematicians, taking a particular stance only when conflicting but well-established traditions require this. This makes FMathL compatible with multiple, possibly conflicting philosophies regarding the meaning of terms outside their intended use.

In particular, the abstract object level defines the FMathL ontology of mathematical objects and makes – unlike traditional foundations – statements, numbers, texts, functions, and sets (possibly) different kinds of objects. In a further reflection process, the implementation level then realizes the (reflected) objects in a particular way, in one of many possible alternatives.

1.4 Subject levels and object levels

And God said, Let there be light, and there was light.

(Genesis 1:3 [38])

In this section, we give an intuitive, informal picture of how the FMathL approach works. In Chapter 2, we make this picture precise at the level of usual mathematical standards of rigor.

In the foundations of mathematics, it is necessary to carefully distinguish between the **subject level** and the **object level**.

People (and machines) may have their subjective views about what a mathematical object, a number, a function, etc. is, as long as they agree on the properties specified in the axioms, and make the same definitions based on these. The subjective views constitute the subject level, whereas the part on which there is agreement, enforced by some standard (in our case by the axioms for a mathematical framework), constitutes the object level. Since the axioms underdetermine the object level, there is much room for subjective variation. But if everything communicated can be reduced to the axioms and the definitions, perfect communication is possible in spite of this variation.

Thus there are different subject levels, namely at least one for each **subject** doing mathematics. (See Section 3.4 for the discussion of an infinite hierarchy of subject levels for the same subject.) Within each subject level, there is a carefully structured domain, the object level, private to each subject and nevertheless public in a certain objective sense to be made precise.

For foundational purposes one mostly works within a fixed subject level referred to as *the* subject level. The corresponding object level is then referred to as *the* object level.

Subject level and object level relate to each other like mind and matter. All concept formation, reasoning, and discussion happens on a subject level like in a mind, or between subject levels like between minds. Like matter by the mind, the object level is accessed only through reference: pointing to something, describing something, a flash of insight triggered by viewing the context of something, etc.. How this may result in objective communication is discussed in Section 3.2.

On the object level, formally defined in Section 2.1, we have an infinite collection of mathematical **objects**, related by a number of intrinsic binary operations; some objects are individuals distinguished by a canonical name. Objects simply are, static, timeless; given together with infinitely large operation tables for the intrinsic operations. The latter is not to be taken literally; it is just a handy intuition for the maps defining each operation. Mathematicians usually only create a tiny finite part of such an operation table.

Objects behave like instances of abstract data structures in computer science (lists, heaps, stacks, etc.): we know nothing about them except what is given through the axioms describing their constructors and methods, and it is up to the implementation how to represent and access them in a memory whose properties are irrelevant for the ordinary user and matter only for a particular implementation.

Objects are combined, manipulated and reasoned about on the subject level: Only subjects can do something with objects, and each subject can manipulate only the objects from its own object level.

In particular, all mathematical language used, all mathematical formulas, and even the axioms and definitions belong to the subject level, although they point to objects on the object level and establish relations between such objects.

For example, carrying out an operation defined on the object level is an activity on the subject level. One looks up the two objects to be combined by an operation in the corresponding operation table and then reads off the result from the table.

Introducing an object into the discussion by saying "Let n be a natural number" is another activity on the subject level. The subject ensures by this activity that the context represented on the subject level contains the observation that n is a natural number that was not present there before; cf. the quotation introducing this section. In this sense, the subject level is the operating system on which the object level (an infinite piece of structured memory) can become alive.

Asserting truth. The concept of truth is handled separately by the concept of contexts. Certain objects of the object levels are characterized as being **statements** (see Section 2.4), capable of being **asserted** as truths.

In common mathematical language, mathematical objects are mentioned in two different ways, either by simple **reference**, or to declare an object to be a **truth**; in the latter case, the object must be a statement for the declaration to make sense. When talking about the first case, we say that we **reference** the object; when talking about the second case, we say that we **assert** the object. In both cases, we **mention** the object by writing an expression for it; cf. Section 2.2 below. In common mathematical language, this language is used explicitly only for emphasis or when necessary to avoid ambiguity, since usually the context together with the form of the object determine uniquely whether an object mentioned is referenced or asserted. For example, when we say, "consider the statement $x = y$" then $x = y$ is mentioned but not asserted, while when we say, "suppose $x = y$" or "we find $x = y$", then $x = y$ is asserted. As part of the common mathematical language, it is therefore assumed to be clear on the subject level when an object is asserted.

A **context** (see NEUMAIER & MARGINEAN [56] and Section 2.5) is a collection of statements considered to be true. What is then actually true is context-dependent: In a given context, something is true iff it is part of the collections of truths in the closure of this context. The **closure** of a context is closed under three extremely simple forms of reasoning – false reflection, and reflection, and equal reflection – that are basic for mathematical reasoning, and in fact enough if one adds the double negation law; possibly also under other principles not specified explicitly. In particular, it is assumed that every closed context contains all instantiations of the assertions in the axioms of a mathematical framework (defined below) and hence all of its consequences.

Mathematicians often change contexts, even within a single line of argument. For them, a context is something that is "known" in the weak sense of being assumed to be true (i.e., to contain all currently relevant truths), often only temporarily for the sake of exploring its consequences. Changes of contexts are effected by various means.

An assertion on the subject level (such as "suppose that $x = y$") changes that context, adding the asserted statement (here "$x = y$") to the old context to form a new one, thereby declaring it to be true. The context remains the same iff the statement was already in the context. The closure remains the same iff the asserted statement was true in the old context; otherwise it changes, sometimes drastically. Reasoning in some context amounts to adding to a context statements that can be derived from statements already in the context by the three reflection rules and their consequences, or by asserting statements that are instantiations of known axioms, propositions, or theorems.

A context is **inconsistent** if its closure contains all statements – including "false is true", the presence of which already ensures that the context is inconsistent. In a given context, a statement is a **contradiciton** if its assertion leads to an inconsistent context.

In principle, to check whether $x = y$, say, holds, one looks up in row x and column y of the operation table of $=$ the result of evaluating this expression, and then checks whether this result or its negation is contained in the closure of the current context. If so, we know that $x = y$ is (currently) true or false. If not, we know nothing about the validity of $x = y$. Of course, whether we can actually perform these activities depends on whether we are able to construct explicitly the relevant part of the operation table and of the context closure. This is not always possible, whence the truth of a statement may be undecidable. In particular, this will often be the case for mathematically unmotivated statements such as POINTWISE = SET + SMALL.

Often, within a mathematical text, the context is augmented, reduced, or changed completely as needed, according to established informal principles. In particular, in indirect proofs and in arguments by cases, extra assumptions are introduced, to be removed again when a goal or a contradiction has been reached. A change of context may even alter the meaning of words and symbols. A change of context may be indicated in mathematical arguments by headings like "Chapter 3" and phrases such as "let ...", "Case 1. ...", "Contradiction. Therefore ...", "We assume ...", "This concludes the proof", "As a preparation, we consider ...", "In this section, we write ...", etc.. Students of mathematics usually learn early in their studies the cues that tell when and how contexts change; in this document we simply assume them known. (See the papers by KOLEV [40] and KUEHLWEIN [45] on the Naproche project for a formal approach to such cues.)

Using the subject level as an informal basis, the object level is carefully specified in a way that, ultimately, everything is checkable by a machine, as far as the theoretical limitations on computability allow. The part of the subject level that is expressible formally on the object level is termed **objective**; everything else is termed **subjective**. Because of the common properties of the object levels of different subjects, and because of the standardardized way of referencing the object level, objective communication is possible about everything objective.

In order to reason objectively on the subject level, one embeds the part of the subject level that is relevant for a particular discussion into the object level, thereby achieving objective communicability. The process achieving this embedding in a manifestly equivalent manner is called **reflection**.

1.5 The subject level

In the present document, the language on the subject level is used in the informally rigorous manner in which mathematicians usually communicate their concepts and theories.

Logic as part of mathematics. The traditional setting for the foundations of mathematics is predicate logic, which is based upon a syntactical approach. Here, a rigid syntax for well-formed formulas defines the object level; the subject level is concerned with the definition, parsing, and manipulation of formulas. Later, the semantics of formulas is specified separately by a corresponding model theory.

In direct contrast, the language on the subject level of FMathL is used in the informally rigorous manner in which mathematicians usually communicate their concepts and theories. The resulting syntax-free abstract approach formulates on the subject level *everything* that happens in mathematics; in particular, all expressions and formulas belong to the subject level. The formulas have an immediate semantics since there is a fixed object level in which the mathematical objects are defined. Later (in another paper [58]), the syntax of admissible formulas and verbal constructs is specified separately by a corresponding formal grammar that reflects standard mathematical conventions.

In short, while the traditional approach tries to reduce mathematics to logic by treating the foundations of mathematics as part of logic, the FMathL approach reduces logic to mathematics by treating logic as part of mathematics.

I consider this more natural since neither the syntactic approach to logic nor model theory, which gives it some semantics, can be rigorously defined without exploiting on the subject level the standard mathematical tools. As in any field of study where conceptual precision is needed, the precision is obtained by phrasing the concepts in the field by means of mathematics. In particular, conceptual clarity for both mathematics and logic is onbtained by basing it on mathematics.

Also, as will be argued in Section 3.2, not the *logical* proof – anyway available only for a tiny fraction of today's mathematical edifice – but the *mathematical* proof is the essence of the reliability of mathematics in the real world.

Requirements for the subject level. From a linguistic point of view, we assume that a **subject level** can interpret the usual informal mathematical language (to the extent used in the axioms, definitions, and proofs in this document) and accepts classical logic and a set theory in which unions of countably many countable sets are countable, and in which quantification over all subsets of a countably infinite set make sense.

Thus, on the subject level, we have available the standard machinery of classical logic including quantifiers, and the usual set-theoretic and algebraic language for describing and arguing with mathematical objects. In particular, we know on the subject level the meaning of concepts such as a subset or a field.

One possible suitable formalization of the language of the subject level would be constituted by ZFC, Zermelo-Fraenkel set theory with the axiom of choice, together with Bourbaki-style abuses of language and notation (though these are both difficult to formalize and difficult to avoid). However, different implementations of the subject level in a language such as illative combinatory logic, a type-theoretic logical framework in a computer system, or category theory, appear possible. Moreover, the reflection process described briefly in Section 3.4 will take the FMathL object level itself as a subject level, within which a reflected object level will be constructed.

However, due to our requirement that subjects understand the informal mathematical language used in this document, the rigorous formalization requires in each case a nontrivial amount of work.

While on the subject level the various implementations may differ, on the object level it is as if they did not, thus achieving behavioral equivalence in terms of the portion of mathematics that can be communicated objectively.

There need not be a definite commitment on the subject level. Indeed, it is likely that each (human or computer) subject doing mathematics has its own, personal subject level, only imperfectly corresponding to one of the formal subject levels mentioned. The important thing is not the subject level itself but the agreement of what can be said in a uniform way about all object levels defined on different subject levels.

In a theory of admissible subject levels, it would be required that the corresponding object levels are equivalent in the sense of allowing precisely the same deductions. In the absence of such a theory, one hopes that a subjective check for the validity of the axioms for a mathematical framework in one's preferred subject levelin one's preferred subject level suffices to achieve this purpose.

Reflection. The semantics of the subject level and all syntactical issues are specified later as part of a well-defined **objectification** (or, in computer science terms, **reification**) process based on systematic **reflection**, which makes objective the properties of the subject level needed for the description of the FMathL mathematical framework.

We shall use on the subject level only constructs that are later (in [57] and [58]) reflected into the object level. Thus, ultimately, each concept will exist both as a concept on the object level and as a corresponding metaconcept on the subject level. In particular, we shall have text that reflects the metatext used to specify the properties required of the object level. This will ensure that we can rigorously reason about the way mathematicians both define concepts and reason about these concepts.

Thus, we follow the usual bootstrapping practice of mathematical teaching, in which one first learns about the fundamental mathematical concepts on a more naive basis before following a course on formal foundations.

Reflection amounts to teaching the part of the subject level relevant for doing mathematics faithfully to a machine that understands only the object level. Everything a mathematician does on the subject level while doing mathematics is then representable objectively on the object level, and therefore can be reproduced by a machine.

Achieving this in a natural way close to actual practice is the primary objective of the FMathL approach to the foundations of mathematics.

1.6 The object level

In the following, we create a specification of the object level, general enough and flexible enough so that it becomes fairly easy to construct a faithful model of the subject level inside the object level. This will be done at a later stage by defining within the object level an algebraic structure called a **mathematical framework** (see the companion paper [57]), whose definition reflects the concepts and axioms for the object level that are specified in the following on the subject level only. In a mathematical framework it is then possible to reflect (within certain limits) upon the subject level via the object level.

This makes the object level objective in the sense that it makes unambiguous communication possible between two subjects whose subject levels implement the object level in different ways, but both satisfying the specifications.

Indeed, one can prove a completeness theorem for mathematical frameworks as defined by the FMathL axioms, so that a statement holds in all mathematical frameworks precisely when the statement is provable. Thus everything **objective**, and nothing else, is **provable**. Since each proof can be represented as a finite piece of text in the common mathematical language, it can be reflected into the object level and then be communicated unambiguously between subjects.

Metaconcepts. Notions usually referred to by the same name may denote different entities on the subject level and on the object level. Thus we need to distinguish between metaobjects (on the subject level) and objects (on the object level), metasets and sets, metaoperations and operations, metatexts and texts, metacountable and countable, metatheorems and theorems, etc.. However, adding the prefix meta to *every* use of a metaconcept would clutter the language too much. We therefore add the prefix **meta** only where it seems necessary to remain unambiguous; the unqualified name refers to the object level or to the subject level depending on whether or not, in the context of its use, a concept with that name is (already) defined on the object level.

For example, the set $\mathbb{O}$ of all objects in the object level is a metaobject, and will be called a metaset once sets are introduced on the object level (which we do in Section 2.14). Similarly, the operation $\in$ on $\mathbb{O}$ is a metaobject, and will be called a metaoperation once operations are introduced on the object level (which we shall do in the companion paper [57]).

A fully unambiguous way of proceeding – necessary (only) for a machine interpretation of FMathL – will be deferred to the specification level of FMathL [58].

Axioms. The main assumptions about the object level are formulated in 27 **axioms**, consecutively labelled by A1–A27. Certain properties within the axioms are labelled by (P1)–(P46). Any mathematical structure satisfying these axioms will be called a **mathematical framework**; see Section 3.1.

The axioms are chosen in a way that, as a mathematician, one can readily convince oneself of their truth in one's private (but publicly educated) view of mathematical concepts. In this way, the familiarity with traditional mathematics implies a familiarity with the consequences of the FMathL axioms, and facilitate trusting the axioms to be consistent. (See Section 3.2 for a further discussion of this topic.)

Our language is informal, appropriate to the subject level within which the object level is defined. In particular, all axioms are formulated in the usual, informal style that mathematicians use. Formalizing the axioms themselves, and formally defining the language for doing so, is part of an additional reflection process that will be discussed in a separate paper (Neumaier et al. [58]) on the FMathL specification level.

In our exposition, the axioms are complemented by comments that explain the assumptions, by definitions that introduce additional notation for their flexible use, and by properties deduced from the axioms. But we only deduce what seems necessary to see that all mathematics can be faithfully represented inside the FMathL mathematical framework. The fully flexible notational surface layer representing whatever mathematicians like to define is more complex, but only exploits properties of the object levels already presented here. Its discussion is therefore also delegated to [58].

In the following, we objectify

- mathematical objects – the material of the object level and how to combine it (Sections 2.1 and 2.2),
- existence – the ontology of the object level (Section 2.3),
- statements – the representation of potential assertions and their consequences (Sections 2.4 and 2.5),
- equality – the representation of abstraction from detail (Section 2.6),
- membership – the concept of belonging to an object (Section 2.7),
- numbers – the core of elementary mathematics (Section 2.8),
- texts – the dominant communication medium of mathematics (Section 2.9),
- application – obtaining values at specific arguments (Section 2.10),
- functions – the representation of transformations (Section 2.11),
- abstraction – the representation of properties defined by expressions (Section 2.12),
- categories – the representation of mathematical structures (Section 2.13),
- sets – the representation of collections (Section 2.14),
- infinity – the representation of "and so on" (Section 3.3),

- countability – the notion of counting objects (Section 3.5),
- quantifiers – the representation of something and everything (Section 3.6).
- paradoxes – the deduction of nonexistence (Section 3.7),

This allows us to objectify (in the companion paper [57]) mathematical frameworks, of which the (reflected) object algebra of the FMathL mathematical framework can be seen as a particular instance. This reflection process is further discussed in Section 3.4 below.

Chapter 2

The axioms

Note that each axiom extends to the end of the paragraph containing the paragraph. The discussion of the axiom begins with a new paragraph.

2.1 Objects

Axiom A1.
$\mathbb{O}$ is a fixed, countably infinite set whose elements are called **objects**.

The **object level** consists of everything defined algebraically within $\mathbb{O}$ in terms of the above operations and distinguished objects. Note that $\mathbb{O}$ itself is not an object but only a metaobject.

Objects are given to the mathematical discourse only by reference; for example, to say that 0 is an object only means that 0 is a label pointing to the object represented by 0. For a discussion of our conventions for talking about objects and their names (abstracted from common mathematical practice) see Section 2.2.

The countability assumption on the subject level has two important objectives.

(i) On the subject level, it gives quantification over all objects a foundationally innocuous, nearly constructive interpretation to which not even KRONECKER [44], who said in 1886, "Die ganzen Zahlen hat der liebe Gott gemacht, alles andere ist Menschenwerk" – that God made the natural numbers, but everything beyond is made by man, would have objected. (For agnostics and atheists: Made by God = isomorphicly given on each subject level.)

(ii) It ensures that it is possible to encode (as usual in mathematics) *all* objects in terms of texts over a suitable finite alphabet. This can be done in many ways; one of them is described and discussed in Section 3.5; it provides an FMATHL version of the Löwenheim-Skolem theorem (LÖWENHEIM [49], SKOLEM [68]), which says that every consistent theory in first order logic has a countable model. This theorem ensures that assuming $\mathbb{O}$ to be (meta-)countable is not a restriction of generality. Since the details of the encoding are irrelevant for the foundations, the present syntax-free abstract setting is sufficient to do everything of interest without cluttering the presentation too early with a discussion of complex syntactical issues on the object level.

But the Löwenheim-Skolem theorem also ensures that if there is some countably infinite model then there is a model whose cardinality is any given infinite cardinal number. Thus there is a multiplicity of possible nonisomorphic implementations of the subject level. The concept of countability on the subject level is therefore necessarily subjective - i.e., implementation-dependent. On the other hand, the intrinsic notion of mathematical concepts, in particular that of countability on the object level is objective, implementation-independent, and hence transferable between different subject levels. Its meaning is the same in all subject levels.

Because of its fundamental importance for clarifying the reflection process, we shall discuss countability on the subject level and the object level again after having the mathematical framework fully defined. We shall find that the countability assumption in Axiom A1 is only a convenience for the subject level, and not an essential constraint. Indeed, it is shown in Section 3.5 that one could do without it, and recover from each noncountable $\mathbb{O}$ a more or less canonical subset of objects satisfying all axioms including countability on the subject level. Thus, for practical purposes, there is no restriction in having assumed countability from the outset.

Distinguished objects. In order to be able to reflect basic concepts from the subject level inside the object level, the presence of certain objects must be postulated.

Axiom A2.
Among the objects in $\mathbb{O}$ are 29 distinct distinguished objects, namely 7 standard objects denoted by the mathematical symbols

$\mathbf{0}$ (false; cf. Axiom A5),
$\mathbf{1}$ (true; cf. Axiom A5),
$\emptyset$ (the empty set; cf. Axiom A10),
0 (zero; cf. Axiom A11),
1 (one; cf. Axiom A11),
$\mathbb{N}$ (the domain of natural numbers; cf. Axiom A11),
$\mathbb{C}$ (the domain of complex numbers; cf. Axiom A11)

and 22 auxiliary objects, denoted by the following words (in small caps to emphasize that they should be regarded as single symbols):

ABS	(the absolute value; cf. Axiom A12),
CHAR	(the set of characters; cf. Axiom A14),
CHOICE	(the choice function; cf. Axiom A20),
COD	(for defining codomains; cf. Axiom A21),
CONST	(for defining constant functions; cf. Axiom A20),
CONJ	(conjugation; cf. Axiom A12),
DOM	(for defining domains; cf. Axiom A21),
FUN	(for defining functions; cf. Axiom A18),
HOM	(for defining homomorphisms; cf. Axiom A22),
ID	(the identity; cf. Axiom A20),
ID_	(for defining local identity; cf. Axiom A21),
INF	(for the infimum of real numbers; cf. Axiom A13),
INTERSECT	(defining arbitrary intersections; cf. Axiom A10),
LOW	(for lower bounds of real numbers; cf. Axiom A13),
MAP	(for creating maps; cf. Axiom A25),
NULL	(the empty text; cf. Axiom A14),
POINTWISE	(for defining expressions; cf. Axiom A20),
SET	(the category of sets; cf. Axiom A23),
SMALL	(for characterizing small sets; cf. Axiom A23),
SUBSETS	(for power sets; cf. Axiom A27),
TEXT	(the set of texts; cf. Axiom A14),
UNION	(defining arbitrary unions; cf. Axiom A10),

Intrinsic operations. Any computer implementation of mathematics must be able to do in a finite number of basic steps every activity that mathematicians perceive as a single step in their daily work. Thus the set of objects manipulated must have corresponding intrinsic operations that account on the abstract level for the activities constituting a single basic step.

On the implementation level, the intrinsic operations provide the basic constructors for recursively constructing new objects from the few distinguished objects, in the same way as one constructs in Peano arithmetic particular natural numbers from the first such number by repeated application of the successor operation. In this sense, all of mathematics is made constructively available in a bottom-up fashion.

The sort of constructivism resulting in FMATHL is different from that traditionally going under the heading of constructive mathematics as done e.g., in BISHOP [12]. There constructing a real number, say, does not mean giving some particular string characterizing the number but being able to give an algorithm that displays the digits of a decimal number expansion one after the other, identifying which one of the digits 0, 1, 2, 3, 4, 5, 6, 7, 8, or 9 it is. Similarly, any other construction must give a definite, unique normal form representation in concrete data of a particular sort that allows to decide trivially whether two objects are equal.

In the FMathL form of constructivism, however, $\textsc{Choice}(\{a, b\})$ – which we will soon recognize as being the mnemonic way of writing $\textsc{Choice} @ (a \sqcup b)$ – is a finite string constructing a definite element of the object $\{a, b\}$; cf. Section 2.7. It is either a or b, but it will never be known objectively whether this element is a or b since there is no rule that would allow to deduce such an identification. Thus one implementation of the FMathL mathematical framework may have $\textsc{Choice}(\{a, b\}) = a$, while another implementation may have $\textsc{Choice}(\{a, b\}) = b$, and a third implementation may not commit itself at all – without any violation of the axioms, and without impairing the contents of the mathematics communicated by the construct.

The decoupling of constructivity and decidability achieved by FMathL in this way – in the spirit of Hilbert and Bourbaki – accounts for its flexibility in handling arbitrary mathematical contents.

Axiom A3.
$\mathbb{O}$ admits 14 binary **intrinsic operations** ω that assign to any two objects $x, y \in \mathbb{O}$ another object $x\,\omega\,y$. The set Ω of intrinsic operations defined on $\mathbb{O}$ is disjoint from $\mathbb{O}$ and consists of the operations

- $\wedge$ (**and**, defining **conjunction**, Section 2.4),
- $=$ (**equal**, defining **equality**, Section 2.6),
- $\in$ (**in** or **belongs to**, defining **membership**, Section 2.7),
- $\sqcup$ (**together with**, defining objects with at most two elements, Section 2.7),
- $+$ (**plus**, defining **addition**, Section 2.8),
- $-$ (**minus**, defining **subtraction**, Section 2.8),
- $*$ (**times**, defining **multiplication**, Section 2.8),
- $/$ (**divide**, defining **division**, Section 2.8),
- $\&$ (**append**, defining **concatenation**, Section 2.9),
- $@$ (**at** or **of**, defining **application**, Section 2.11),
- $\square$ (**image of**, defining images of sets, Section 2.10),
- $|$ (**selection**, defining objects with elements of selected properties, Section 2.10),
- $\to$ (**to**, characterizing arrows, Section 2.13),
- $\diamond$ (**following**, defining the **arrow product**, Section 2.13),

The equality sign is used as a binary operation symbol rather than as a relation sign; we use $x \equiv y$ to express that two objects x and y are **identical** as elements of $\mathbb{O}$.

The meaning of the operations and distinguished objects, and the properties required of them to give them this meaning, will be specified in subsequent sections, which give our setting a semantic interpretation in terms of standard mathematical practice. Note that there are many objects that can be constructed with the intrinsic operations but are useless for doing mathemartics, just as there are many chess positions that will never arise in actual play.

The few operations with noncanonical symbols, &, @ , $\square$, $|$, $\diamond$, and $\sqcup$ also correspond to well-known mathematical operations usually expressed in a different syntactic form. The standard mathematical notation for @ , $\square$, $|$, and $\sqcup$ will be given in Section 2.2, and will be used later except on their first occurrence and where absolutely necessary. The mnemonic symbols & for the concatenation of text and $\diamond$ for the composition of arrows were chosen to avoid unnecessary ambiguity.

The object level consists of everything defined algebraically within $\mathbb{O}$ in terms of the above operations and distinguished objects.

2.2 Variables and expressions

We now look more closely at the way objects are referenced on the subject level. This is done by means of expressions, special texts on the subject level given by well-formed formulas involving only the names of distinguished objects and the symbols for the primitive binary operations.

An **expression** of **length** $L = 1$ is a **name** denoting either one of the distinguished objects listed in A2 or – if not in this list – a **variable**, i.e., a named hole for objects. Mathematicians are quite liberal in what they consider as a name for a variable; so FMathL makes no particular assumptions about it.

An **expression** of (integral) **length** $L > 1$ is a text of the form $(E\,\omega\,E')$, where ω is the symbol for an intrinsic operation, and E and E' are expressions whose lengths sum to L. ω is called the **final operation** of E, its **subexpressions** are E, E', and their subexpressions. Expressions of length 1 have no final operation and no subexpressions.

Note that this happens on the subject level; names and hence variables and expressions are metaobjects, not objects. Even when we say "let x be an object", x is just a name pointing to the object introduced by "let"; similarly, $\mathbb{C}$, say, is just a name pointing to the object characterized by Axiom A11 below. Objects themselves are not accessible directly in mathematical discourse. This is necessary since different subjects may have completely different implementations of the "same" object.

Note that $\equiv$ is not an intrinsic operation; therefore $x \equiv y$ is not an expression. A formula such as $x \equiv y$ involving an operation neither intrinsic nor defined on the object level is called a **metaexpression** if it is otherwise formed according the above prescription.

The substitution of objects for variables in an expression E is unrestricted. Such a **substitution** is determined by an **assignment** of objects to each variable used in E, followed by a recursive evaluation of all intrinsic (or derived) operations. It results in an object $E()$, the **result** of **evaluating** E, depending on the assignment. Usually, the assignment is apparent from the informal context. A **definition by abbreviation** is a formula (metaexpression) of the form

$$a := E$$

with an expression E all of whose variables are assigned in the informal context. In actual usage, E is of course replaced by the corresponding explicit expression. This definition expresses that a is the unique object $E()$ obtained by substituting for the the names in E the corresponding objects known from the informal context, and evaluating the operations with which the subexpressions are formed, giving **intermediate results**. In particular, $a \equiv E()$.

The convention in common mathematical language, and hence also here, is that a formula which looks like an expression is always treated as evaluated, i.e., as reference to an object, except when the context requires the interpretation as an expression. This is the case when the context mentions the word expression (or formula, string, symbol, etc.) denoting the formula in question and in constructs that involve bound variables, i.e., in λ-notation, in the curly bracket notation for selection and images, and in quantified formulas.

For example, $\mathbb{C}$ is usually regarded as an object, namely (by Axiom A11 below) the field of complex numbers, but occasionally not, e.g., in a context like "to be able to express the FMathL axioms, we need on the subject level the character $\mathbb{C}$".

Abbreviations are the simplest form of definition. Later, we shall meet other kinds of definitions for which the same notation := is used: definition by abstraction in Section 2.12, and self-referential definitions in Section 3.7.

Concerning possible ambiguities on the informal subject level, we simply note that mathematical notation has always been ambiguous at times, in order to achieve brevity and easy comprehensibility in the typical usage. Teaching mathematics to the computer requires an awareness of all such ambiguities, and how to resolve it. *Good* mathematical writing uses ambiguous formulations only when these can be disambiguated easily from the informal context, making this resolution a manageable task.

Thus, we do not exclude ambiguities from the subject level but only ensure their unique interpretability in their informal context.

To match traditional notation, we write expressions more liberally, with rules that allow one to replace the liberal notation by one strictly conforming to the above definition. In particular, we write

$$\begin{array}{lll} f(x) & \text{for} & f \mathbin{@} x, & (\textbf{evaluation}) \\ f(x,y) & \text{for} & f(x)(y), & (\textbf{currying}) \end{array}$$

and similarly for $f(x,y,z)$, etc.,

$$\begin{array}{llll} |x| & \text{for} & \mathrm{A{\scriptsize BS}}(x), & (\textbf{absolute value}) \\ \overline{x} & \text{for} & \mathrm{C{\scriptsize ONJ}}(x), & (\textbf{conjugation}) \\ xy & \text{for} & x * y, & (\textbf{multiplication}) \\ \mathrm{I{\scriptsize D}}_A & \text{for} & \mathrm{I{\scriptsize D}}\_(A), & (\textbf{identity on } A) \\ \{x,y\} & \text{for} & x \sqcup y, & (\textbf{unordered pair}) \\ f[Z] & \text{for} & f \square Z, & (\textbf{image}) \\ \{f(x) \mid x \in Z\} & \text{for} & f \square Z, & (\textbf{image}) \\ \{x \in Z \mid f(x)\} & \text{for} & Z \mid f. & (\textbf{specification}) \end{array}$$

Each of these notations will be explained again on first usage.

Note that in the last two formulas, the letter x is a dummy letter (usually referred to as a bound variable) that is not part of the algebraic structure; hence it can be replaced by any other letter distinct from Z and f (and other letters which would be confusing in a particular context) without altering the meaning; e.g., $\{x \in Z \mid f(x)\}$ and $\{z \in Z \mid f(z)\}$ are just different (in the context usually "telling") ways of referring to $Z \,|\, f$. The only role of the bound variable is to connect the formal algebra in $\mathbb{O}$ to the traditional notation on the subject level, where we communicate many formulas using bound variables rather than using algebraic operations.

We shall meet later numerous other defined notations including new operation symbols such as $\neg$, $\not\in$, etc., which may be used in forming expressions, but which could in principle be removed by substitution.

To be able to write expressions without unnecessary parentheses, we adopt the convention that evaluation written as $f(x)$ binds strongest, invisible multiplication binds stronger than explicit operations, the operator $\neg$ binds stronger than all explicit binary operations, $/$ binds stronger than $*$, which binds stronger than $+$ and $-$, which bind stronger than $\in$, $\not\in$, and $\ni$, which bind stronger than the set operations $\cup$, $\cap$, and $\setminus$ introduced later, which bind stronger than $\wedge$, $=$ and $\neq$, which bind stronger than $\Rightarrow$ and $\Leftrightarrow$, which bind stronger than $\equiv$. Operations of the same priority are applied from left to right. Only deviations from these rules need to be marked by parentheses or by additional locally specified priority rules; however, pairs of parentheses may also be added for the sake of clarity or emphasis. For example,

$$1/2x = 1/(2x), \quad 1/2 \cdot x/y = (1/2)(x/y),$$

$$a + 1/a = a + (1/a), \quad a - b - c = (a - b) - c.$$

Local priority rules, where the priority is deduced from the context, are particularly meaningful for the binding of $\wedge$, $=$ and $\neq$. In equational logic, one would want $\wedge$ to bind stronger than $=$ and $\neq$ (the convention used in [56]), while for mathematics in general, one would like $=$ and $\neq$ to bind stronger than $\wedge$ (the convention used in the following), and uses the comma as a replacement for $\wedge$ that binds weaker than $=$ and $\neq$.

2.3 Existence

Mathematicians often reason with objects such as "the largest prime number" whose existence is in doubt, in order to prove them either existent or nonexistent. A famous example is the proof by Feit & Thompson [25] that there is no noncyclic finite simple group of odd order, where much of the argument is about "the smallest noncyclic finite simple group of odd order", a nonexistent object that at first seems quite respectable but, after several hundred pages of reasoning based upon its properties, is shown not to exist. Other, existing objects like the finite simple group called the monster, were later constructed by similar methods, first finding property after property of this hypothetical group until its character table and some important subgroups were known. From these, such a group was constructed by Fischer & Griess [28] after additional work.

Thus there must be a stronger concept of existence of objects that goes beyond the obvious fact that an object exists as an element of $\mathbb{O}$ (which is necessary to reason formally about its properties).

Objects that do not exist in this stronger sense are traditionally called nominal; they have a name (nomen) which defines the object but no real existence beyond that name. The term "nominal" goes back at least to William of Ockham [60] around 1320; the distinction between real existence (of primary entities = "protoi ousiai") and nominal existence (of secondary entities = "deuterai ousiai") even to Aristotle [2, 3] around 350 B.C..

According to Ockham, a definition (i.e. a simple or composite name for an object) may be real ("quid rei"), signifying an absolute thing (e.g., a tree, whiteness, taste, or an angel) or nominal ("quid nominis"), signifying a thing indirectly (e.g., white or length). Thus, for Ockham, an object such as a tree (corresponding to urelements in a set theory) would have real existence, while an object such as length (a function in a set theory) would have nominal existence only, and get real existence only when evaluated ("length of the tree").

Modern mathematics has a more liberal view of existence but still benefits from the distinction between strong, real existence and weak, nominal existence. Nominal mathematical objects (such as $0/0$ in arithmetic) are also useful for writing down formulas that do not make sense for all possible arguments (such as in arithmetic x/y for $x = y = 0$), and to be able to reason about paradoxes (see the discussion of Axiom A5 below, and Section 3.7).

The dual meaning of the term "exists" is naturally accommodated in the FMathL setting by assigning the weaker sense (metaexist = have existence as real *or* nominal essence) to the subject level and the stronger sense (exist = have existence as real essence only) to the object level: All objects metaexist, but some of them are nominal and have no "real" existence. The FMathL propagation rules, however, are adapted to current mathematical practice, and hence quite different from those suggested by Ockham.

Axiom A4.
We say the object x **exists** if $x\,\omega\,y$ or $y\,\omega\,x$ holds for some object y and some intrinsic operation ω; otherwise the object x is called **nominal**. All distinguished objects described in Axiom A2 exist.

Thus the result of operations involving at least one nominal object is always a nominal object. This restrictive rule is chosen so that logical arguments with nominal objects will not give rise to inconsistencies. While nominal (i.e., nonexisting) objects can be named and manipulated on the subject level, they have a very inferior ontological status compared to existing objects. Nontrivial logical arguments with nominal objects are very limited, essentially being restricted to their occurrence in expressions involving bound variables, see Section 2.12 below.

The value of nominal objects lies in the fact that we can form objects without restriction, without having to bother about their existence. We can then investigate their properties under the assumption that the object exists, and if this leads to a contradiction, conclude that the object must in fact have been nominal only. Thus nominal objects are typically formed only to discard them once proved nominal.

Axiom A4 implies that if x and y are objects for which $x\,\omega\,y$ exists for some primitive operation ω then x and y exist. Thus one can infer from the existence of the result of an expression the existence of all its intermediate results. For example, the assertion of $x \in Z$ entails the existence of both x and Z. The FMathL axioms usually imply the existence of interesting objects in this indirect way.

On the other hand, one can usually *not* infer the existence of $x\,\omega\,y$ from that of x and y, though this can be proved from the axioms under suitable restrictions. In particular, the substitution of objects for variables in an expression E is not always meaningful since the result $E()$ of evaluating E may be nominal. We say that the expression E is **defined** for an assignment if $E()$ exists.

2.4 Statements

Now we reflect the concept of a statement. Statements are those objects on the object level that can be meaningfully asserted on the subject level; the assertion itself is not a statement but only a metastatement. Asserting a statement means declaring it to be true; unasserted statements need not have a definite truth value "true" or "false".

FMathL writes **false** as $\mathbf{0}$ and **true** as $\mathbf{1}$. The statement $\mathbf{1}$ is the unconditionally true statement, and $\mathbf{0}$ is the unconditionally false statement.

FMathL also identifies equality of statements with their logical equivalence. (Equality in general will be discussed in Section 2.6.) Informally, the assertion that a statement is true is the same as the assertion of the statement itself. This motivates the following axiom.

Axiom A5.
We say that the object x is a **statement** if x exists and $(x = \mathbf{1}) = x$. The objects $\mathbf{0}$ (**false**), $\mathbf{1}$ (**true**) are statements. If the objects x and y exist then $x = y$ is a statement. If x and y are statements then $x \wedge y$ is a statement; otherwise $x \wedge y$ is nominal.

According to our conventions on terminology, we need to refer now, after this axiom, to an assertion on the subject level as a **metastatement** rather than as a statement. For example, all axioms are true metastatements.

Note that we may make use of $=$ in this axiom, since we know already that $x = \mathbf{1}$ and $(x = \mathbf{1}) = x$ are objects, although their properties are specified only in Section 2.7 below. Analogous remarks apply to other usages of operations explained only later in the text.

For existing objects x, the object $x = \mathbf{1}$ is a statement, and – consistent with the definition of a statement – we take the assertion of x to stand for the assertion of the statement $x = \mathbf{1}$. On the other hand, a nominal object x cannot be meaningfully asserted since $x = \mathbf{1}$ is again nominal, hence not a statement.

Thus what is usually called a **meaningless statement** is not a statement in the FMathL sense but a nominal object whose assertion is meaningless. This gives the resulting logic of assertions a 3-valuedness on the subject level, with truth values “true”, “false”, and “meaningless”. Because of Axiom A4, the latter propagates through all operations in the fashion of the weak 3-valued logic of Kleene [41]. (On the object level, there are many more truth values – one for every nominal object, and one for every equivalence class of equal statements. Because many statements are undecidable, there are many such equivalence classes.)

More generally, it follows from Axiom A4 that the assertion of an expression implies the existence of all objects from which the expression is built. In particular, *one cannot meaningfully assert any expression involving a nominal object.* This drastically restricts the handling of nominal objects, and eliminates contradictions from paradoxical definitions; see Section 3.7.

Thus, as usual in mathematics, we may agree that, outside of the defining axioms A1–A27 (which we specify with special care), all formulas involving primitive (or derived) operations implicitly assume that the objects involved and all implied intermediate results exist. Exceptions from this rule are always indicated explicitly.

The assertion of $x = x$ says precisely that x exists, and the assertion of $\neg(x = x)$

is contradictory when meaningful. On the other hand, our discussion implies that it is impossible to write down an expression in x that holds precisely when x is nominal, or when x is a particular nominal object. Thus nominality can be handled *only* on the subject level.

Note that equating the assertion of $x = \mathbf{1}$ with that of x implies that the identity mapping on statements is almost a truth functional in the sense of TARSKI [69], whose undefinability theorem asserts that no sufficiently rich interpreted language can represent its own semantics. In the classification by LEITGEB [48] of properties a theory of truth should have but cannot, what fails in FMATHL is the first desirable property, namely that truth is an (everywhere defined) predicate. In FMATHL, this failure manifests itself in the presence of nominal objects representing apparent assertions.

By Axiom A5, $=$ and $\wedge$ are logical operations on statements. Note that a test $x = n$ for equality between an arbitrary object x and a nominal object n results in a nominal object that is *not* a statement. Similarly, as there is no axiom asserting anything about $x \wedge y$ if neither x nor y is a statement, conjunction is useful (and used) only for statements, and is completely indeterminate otherwise. As already mentioned in the introduction, such a lack of complete determination is typical for the FMATHL approach to the foundations.

The operations $\Leftrightarrow$ (**iff**), $\Rightarrow$ (**implies**), $\neg$ (**not**), and $\vee$ (**or**) are defined in terms of the postulated ones by writing, for statements x and y,

$$\begin{array}{lll l}
x \Leftrightarrow y & \text{for} & x = y, & (\textbf{equivalence}) \\
x \Rightarrow y & \text{for} & x \wedge y = x, & (\textbf{implication}) \\
\neg x & \text{for} & x = \mathbf{0}, & (\textbf{negation}) \\
x \vee y & \text{for} & \neg x \wedge \neg y = \mathbf{0}. & (\textbf{disjunction})
\end{array}$$

Like $\wedge$, the result of these operations is taken as nominal if x or y is not a statement. A similar remark applies to all later definitions that are restricted in their arguments.

Propositional logic. All logical operations will get their usual meaning through the reflection rules introduced in Section 2.5.

If x is a statement, asserting $\neg x$ means declaring x to be false. To get classical reasoning, we postulate the following axiom. (See the discussion in Section 3.1 for a possible variant based on intuitionistic logic.)

Axiom A6.
For any statement x,
(P1) $\neg\mathbf{1} = \mathbf{0}, \quad \neg\neg x = x.$

The usual laws of propositional logic for the logical operations follow from these reflection principles together with this axiom.

2.5 Contexts

The axioms so far imply that all existing unconditionally true statements are equal, although they need not be identical as elements of $\mathbb{O}$, and all existing unconditionally false statements are equal, too.

To discuss the validity of arguments about statements we need to formalize a bit of the subject level (still inside the subject level) and how it is reflected on the object level. We refer to any (meta)set of statements as a **context**. Σ denotes the context consisting of all statements.

All this happens on the subject level; contexts are not sets in the object level sense of Section 2.14. Like $\equiv$, they are meaningful only on the subject level.

Axiom A7.
There is a closure (meta)operation that assigns to each context Γ another context $\overline{\Gamma}$, its **closure**, such that, for all contexts Γ, Δ,

$$\Gamma \subseteq \overline{\Gamma} = \overline{\overline{\Gamma}}, \tag{2.1}$$

$$\Gamma \subseteq \Delta \quad \text{implies} \quad \overline{\Gamma} \subseteq \overline{\Delta}, \tag{2.2}$$

and the **reflection rules**

$$\mathbf{0} \in \overline{\Gamma} \quad \text{iff} \quad \overline{\Gamma} = \Sigma, \quad (\textbf{false reflection}) \tag{2.3}$$

$$x \wedge y \in \overline{\Gamma} \quad \text{iff} \quad x, y \in \overline{\Gamma}, \quad (\textbf{and reflection}) \tag{2.4}$$

$$(x = y) \in \overline{\Gamma} \quad \text{iff} \quad \overline{\Gamma \cup \{x\}} = \overline{\Gamma \cup \{y\}}, \quad (\textbf{equal reflection}) \tag{2.5}$$

hold for all contexts Γ and arbitrary statements x and y.

The way the closure is implemented on the subject level affects the notion of truth. (One trivial implementation would make Σ the closure of every context – but then there is no difference between true and false, and all contexts are inconsistent.)

Informally, $\overline{\Gamma}$ is the set of all assertions that hold (for whatever reasons) in every possible world (whatever this is) in which all statements in Γ hold. The uncertainty in the parenthetical interpretation accounts for the subjective part of the notions of truth and possibility; the reflection principle itself determines what is objective about these.

False reflection says that in a context where a false statement holds, every statement holds, so that the context is **inconsistent**. And reflection says that two statements hold in some context iff their conjunction holds. Equal reflection says that two statements are equivalent in some context iff in the context where one of the statements is asserted exactly the same statements are true as in the context where the other statement is asserted.

These **reflection principles** – which are usually phrased (in a syntactical exposition of logical systems) as pairs of **natural deduction rules**, using nonmathematical symbols specific to logic – appear to be the minimal principles needed for an adequate process of cumulative monotonic logical reasoning, fundamental for doing mathematics.

No assumption is made on how to determine an appropriate context, how (or whether) truth can be established in this context, or on how truth and provability are related. It suffices that the notion of truth is context-dependent and that the truth of a statement x in a given context Γ amounts to belonging to the closure $\overline{\Gamma}$, in a way such that the above properties hold.

In the terminology of NEUMAIER & MARGINEAN [56], where one can find a thorough discussion of Axiom A7 and its consequences, the axiom says that Σ carries the structure of a classical context logic. In particular,

$$\mathbf{1} \in \overline{\Gamma}, \quad (\textbf{true reflection}) \tag{2.6}$$

$$(x \Rightarrow y) \in \overline{\Gamma} \quad \text{iff} \quad y \in \overline{\Gamma \cup \{x\}}, \quad (\textbf{imply reflection}) \tag{2.7}$$

$$x, (x \Rightarrow y) \in \overline{\Gamma} \quad \text{implies} \quad y \in \overline{\Gamma} \quad (\textbf{modus ponens}) \tag{2.8}$$

hold for all contexts Γ. It is shown in [56] that all the usual rules from classical propositional logic are valid; in particular, $\mathbf{1} \neq \mathbf{0}$ (which even holds when $\mathbf{0} = \mathbf{1}$), and we can substitute equal (i.e., logically equivalent) statements for each other without changing the validity of an assertion.

Note that the formal meaning of the notion of a statement does not coincide with the informal meaning in ordinary language. This can be seen from an analysis of the **liar paradox** (EPIMENIDES [21]) from ca. 600 B.C., which concerns the interpretation of the putative object

$$\text{L:=“This statement is false.”} \tag{2.9}$$

Suppose that “This statement” refers to L (as suggested by the definition since, naively, the quoted sentence is a statement). Then L is a statement satisfying $L = (L = \mathbf{0})$. As a statement, L exists, and we can apply the rules of classical logic and obtain $L = \neg L$, hence $L = L \wedge \neg L = \mathbf{0}$, hence $L = \mathbf{0}$, hence $\mathbf{0} = L = \neg L = \neg \mathbf{0} = \mathbf{1}$, contradiction. Therefore “This statement” cannot formally refer to L, and the paradox is resolved. (L may still be a meaningful statement, such as in “Consider the statement $\mathbf{1} = \mathbf{0}$. This statement is false.” Only the reference to itself is provably meaningless.)

2.6 Equality

We now reflect the properties of equality. Informally, equality is a somewhat ambiguous concept; it sometimes refers to identity, but more often refers to identity up to irrelevant details or accidental properties, and thus entails an abstraction process. For example, we typically regard two instances of the same character as being equal if their name, case, and (only sometimes) font is the same, no matter which size they have, whether they are printed or handwritten, and even though any two characters "a" differ in their position.

We capture this ambiguity by specifying equality through its properties rather than through a reference to the identity of objects. Thus it is important to distinguish between identity $\equiv$ on the subject level and equality $=$ on the object level.

The traditional properties of equality fall into two groups: One characterizing the fact that equality behaves like identity in the important aspects, while allowing differences in irrelevant details.

Axiom A8.
For arbitrary existing objects x, y, and z,
(P2) $x = x$;
(P3) $(x = y) = (y = x)$;
(P4) if $x = y$ and $y = z$ then $x = z$.

This axiom expresses that equality of objects is an equivalence relation. In particular, $x = y$ if $x \equiv y$; but the converse is generally false. Note that by A4, $x = x$ is nominal (rather than true) when x is nominal!

We write

$$\begin{array}{lll} x \neq y & \text{for} & \neg(x = y), \\ x = y = z & \text{for} & (x = y) \wedge (y = z), \end{array}$$

with a similar interpretation for $x = y = z = w$, etc.. Note that $x = y = z$, $(x = y) = z$, and $x = (y = z)$ have three different meanings.

The other group of properties of equality concerns the ability to substitute equal objects in formulas.

Axiom A9.
For arbitrary objects x, y, u, and v, and arbitrary intrinsic operations ω,
(P5) $x = u$ implies $x\,\omega\,y = u\,\omega\,y$;
(P6) $y = v$ implies $x\,\omega\,y = x\,\omega\,v$.

Note that $x = u$ (resp. $y = v$) implies that x and u (resp. y and v) exist. Thus this axiom says that the substitution of existing equals in expressions formed with intrinsic operations does not change equalities. Note that the metaoperation $\equiv$ is, by design, not an intrinsic operation; and indeed, the substitution of existing equals in formulas involving $\equiv$ is generally not valid. On the other hand, the substitution of objects for names representing free variables in expressions does not require the existence of the respective objects, except when these formulas are asserted to hold.

2.7 Membership

We now reflect membership. The $\in$ operation is suggestive of an intuitive interpretation of membership in sets or classes. However, FMathL takes a different point of view, better reflecting current mathematical practice.

In ZFC, all objects are sets, and $\in$ is an everywhere defined membership relation. In mathematics, sets have a special property called extensionality (see Section 2.14 below) that makes them differ from arbitrary objects (if these exist): Any two sets with the same elements are equal.

If every object were a set then the only object without elements would be the empty set. However, apart from set theory purists, mathematicians generally think of characters and numbers as being objects not containing any elements. At the very least, questions such as whether or not $1 \in \sqrt{2}$ are not considered to be relevant and are nowhere systematically discussed; indeed, in the traditional set-theoretic foundations, $1 \in \sqrt{2}$ is true or false depending on the details used in the construction of a set-theoretic model of the real numbers, within a fixed axiomatic system for set theory (such as ZFC). Thus the set-theoretic properties of real numbers appear to be accidental byproducts of the construction rather than intrinsic properties of numbers. Similar accidental properties result from different set-theoretic models of the natural numbers; see Benacerraf [11].

Set theory thus appears more like an implementation language (with multiple equivalent implementations) for mathematical concepts rather than as an ontology (which would say what numbers are). (This implementation aspect has been emphasized, e.g., by Forster [29].)

There are many other cases where objects used routinely in mathematics to the right of an $\in$ relation violate extensionality. For example, we write $\sqrt{2} \in \mathbb{R}$. But in the traditional set-theoretical approach to numbers, $\mathbb{R}$ – the ordered field of real numbers – is, strictly speaking, an ordered pair consisting of a field and an ordering relation on it. Now in ZFC, an ordered pair is a set with only two elements. Thus, considered canonically as a set, $\mathbb{R}$ contains only two elements, none of which is $\sqrt{2}$. Expressed in a different way: Is $\mathbb{R}$ an uncountable set, a field, or an ordered field? It is conventionally regarded as any of these but can, in ZFC, be only one of them.

Such formally inconsistent multiple specifications are very frequent in a strictly set-based approach to mathematics, present even in the most rigorous treatments of algebra. Treating it simply as a convenient "abuse of notation" constitutes a serious handicap for computerizing mathematics and must therefore be avoided in a good framework of mathematics for the computer. To resolve such issues cleanly, one needs a framework that allows the set of real numbers and the ordered field of real numbers (which are two distinct objects) to contain exactly the same elements.

In FMathL, we therefore separate extensivity (taken to be specific for sets) from membership (meaningful for arbitrary existing objects). An object x may belong to another object Z, i.e., may satisfy $x \in Z$, without Z having to be a set, and there may be many objects (e.g., numbers or pairs) in which there is no object.

Axiom A10.
If the objects x and Z exist then $x \in Z$ is a statement. No object x satisfies $x \in \emptyset$. The object $x \sqcup y$ (**together with**) exists iff x and y exist, and satisfies

(P7) $z \in x \sqcup y$ iff $z = x$ or $z = y$.

The object $\text{UNION}(A)$ exists iff A exists, and

(P8) $x \in \text{UNION}(A)$ iff $x \in B$ for some $B \in A$.

The object $\text{INTERSECT}(A)$ exists iff A contains some object, in which case

(P9) $x \in \text{INTERSECT}(A)$ iff $x \in B$ for all $B \in A$.

The assertion of $x \in Z$ reflects membership of the object x in the object Z. We say that x is an **element** of Z if $x \in Z$. Thus the object $\emptyset$ has no elements. This object will be required in Axiom A24 to be a set, and is therefore called the **empty set**.

It is customary in mathematics to talk about the object $x \in Z$, meaning the object x with $x \in Z$. We typically (but not always) use lower case letters for objects occurring on the left of $\in$, and upper case letters for objects occurring on the right of $\in$. We write

$$\begin{array}{lll} x \notin Z & \text{for} & \neg(x \in Z), \\ Z \ni x & \text{for} & x \in Z, \\ x, y \in Z & \text{for} & (x \in Z) \wedge (y \in Z), \end{array}$$

with a similar interpretation for $x, y, z \in Z$, etc.. We also write for objects x, y, and z,

$$\begin{array}{llll} \{x\} & \text{for} & x \sqcup x, & \\ \{x, y\} & \text{for} & x \sqcup y, & \\ \{x, y, z\} & \text{for} & \{x, y\} \cup \{z\}, & (\textbf{enumerated object}) \end{array}$$

and similarly for enumerated objects with more entries, such as $\{x, y, z, w\}$. Later, these will be recognized to be sets if all their entries exist. If x is nominal then $\{x\}$, $\{x, y\}$, etc., are nominal objects and no sets. If x and y exist then $z \in \{x\}$ iff $z = x$, and $z \in \{x, y\}$ iff $z = x$ or $z = y$, etc..

We also write for objects X, Y, and Z,

$$\begin{array}{llll} X \cup Y & \text{for} & \text{UNION}(X \sqcup Y), & (\textbf{union}) \\ X \cap Y & \text{for} & \text{INTERSECT}(X \sqcup Y), & (\textbf{intersection}) \\ X \subseteq Y & \text{for} & X \cap Y = X, & \\ X \supseteq Y & \text{for} & X \cap Y = Y, & \\ X, Y \subseteq Z & \text{for} & X \subseteq Z \wedge Y \subseteq Z, & \end{array}$$

etc.. Clearly,

$x \in X \cup Y$ iff $x \in X$ or $x \in Y$;

$x \in X \cap Y$ iff $x \in X$ and $x \in Y$.

(The definition of complements needs abstraction; see Section 2.12 below.)

$X \subseteq Y$ iff $Y \supseteq X$ iff $x \in X$ implies $x \in Y$.
$\subseteq$ is verbalized as **part of** or **contained in** (defining **containment**). $\supseteq$ is verbalized (like $\ni$) as **contains**, causing an ambiguity on the verbal level that must be resolved from the context. X and Y are called **disjoint** if $X \cap Y$ contains no object.

2.8 Numbers

We now reflect the core concept of elementary mathematics, that of a number. Note that the metaconcept of a field is available in common mathematical language, hence on the subject level.

Axiom A11.
The object $\mathbb{C}$ is called the **domain of complex numbers**. We say that the object x is a (complex) **number** if $x \in \mathbb{C}$. With $=$ as equality, 0 as **zero** and 1 as **one**, the operations $+$, $-$, $*$ restricted to $\mathbb{C}$, and $/$ restricted to $\mathbb{C}$ and a nonzero denominator, $\mathbb{C}$ is a field. A number contained in every object Z satisfying $1 \in Z$ and $n + 1 \in Z$ for every number $n \in Z$ is called a **natural number**. The object $\mathbb{N}$ is called the **domain of natural numbers** and satisfies
(P10) $n \in \mathbb{N}$ iff n is a natural number.

In particular, every number exists, and the induction principle holds for families of statements parameterized by a natural number. Nonarithmetic questions about numbers such as whether or not $1 \in 2$ are undecidable with the FMathL axioms; thus $1 \in 2$ may be true, false, or meaningless depending on the implementation, so that none of these is an objective property of numbers.

Note that $\mathbb{C}$ is a field with the equality defined in $\mathbb{O}$ but *not* with identity as equality! The field of complex metanumbers, whose equality is identity, cannot be contained in $\mathbb{O}$ since $\mathbb{O}$ is countable on the subject level. (See Section 3.5 for a further discussion of counterintuitive countability properties.)

We do not regard 0 as a natural number; how un-natural it is is proved by the historical fact that the number zero was introduced (in India about 900 B.C.) many centuries after natural numbers were in regular use (in Babylonia about 1500 B.C.). (On the other hand, FMathL will ultimately provide style options, such that everything represented in FMathL can be printed in an external, human-readable output format in which different conventions (e.g., that $\mathbb{N}$ starts at 0) can be accommodated without resulting logical flaws.)

We write

$$\begin{array}{rcll} 2 & \text{for} & 1+1, & (\textbf{two}) \\ 3 & \text{for} & 2+1, & (\textbf{three}) \\ 4 & \text{for} & 3+1, & (\textbf{four}) \\ 5 & \text{for} & 4+1, & (\textbf{five}) \\ 6 & \text{for} & 5+1, & (\textbf{six}) \\ 7 & \text{for} & 6+1, & (\textbf{seven}) \\ 8 & \text{for} & 7+1, & (\textbf{eight}) \\ 9 & \text{for} & 8+1, & (\textbf{nine}) \\ 10 & \text{for} & 9+1. & (\textbf{ten}) \end{array}$$

$0, 1, 2, 3, 4, 5, 6, 7, 8, 9$ are called the (decimal) **digits**.

Note that numbers are assumed to form a field, a concept – currently defined only on the subject level – that abbreviates a number of well-known conditions on the arithmetic operations. As usual in fields, we write

$$\begin{array}{lcl} -x & \text{for} & 0-x, \\ xy & \text{for} & x*y, \\ x^2 & \text{for} & x*x. \end{array}$$

Those concerned about the fact that not all p-adic numbers (or not all ordinal numbers, etc.) are numbers in the present sense are reminded of the well-known facts that a random number is not a number but a measurable function, a quantum group is not a group but an algebra, and that not every skew field is a field. The specification version of FMathL [58] will therefore not require that a composite name such as "quantum group" (or "skew field" or "ordinal number") is an instance of the concept denoted by the noun involved ("group" or "field" or "number").

Rather than postulating an imaginary unit (what is special about the equation $i^2 + 1 = 0$ rather than, say $j^2 + j + 1 = 0$?), we postulate properties of the associated formation of conjugation and absolute value, which are important for the coordinate-free work with complex numbers. Using terminology introduced in later sections, the imaginary unit can then be defined as $i := \text{Choice}(\{x \in \mathbb{C} \mid x^2 = -1\}$.

Axiom A12.
If x is a number then

$$\overline{x} := \text{Conj}(x)$$

and

$$|x| := \text{Abs}(x)$$

are numbers. For any two numbers x and y,

(P11) $\overline{\overline{x}} = x$,
(P12) $\overline{x+y} = \overline{x} + \overline{y}$;
(P13) $\overline{xy} = \overline{x}\,\overline{y}$;

(P14) $|-x| = |x|$;
(P15) $x\overline{x} = |x|^2$.
A number x is called **real** if $x = \overline{x}$ and **positive** if $|x| = x \neq 0$. Not all numbers are real. For any nonzero number x, the number $|x|$ is positive. If x and y are positive numbers then $x + y$ and xy are positive.

That not all numbers are real must be assumed in order to be able to deduce the solvability of the equation $x^2 = -1$.

For real numbers x and y, we write

$$\begin{array}{lll} x \le y & \text{for} & |y - x| = y - x, \qquad \textbf{(less or equal)} \\ x \le y \le z & \text{for} & x \le y \wedge y \le z, \end{array}$$

and similarly for $x = y \le z$, etc..

Axiom A13.
For all numbers s and arbitrary existing objects Z,
(P16) $s \in \text{Low}(Z)$ iff s is a real number with $s \le x$ for all real numbers $x \in Z$;
(P17) $s \in \text{Low}(Z)$ implies $s \le \text{Inf}(Z)$ and $\text{Inf}(Z) \in \text{Low}(Z)$.

This axiom, which uses the convention to write $f(x)$ for $f\,@\,x$, embodies the completeness of the real numbers as a linearly ordered field. Informally, $\text{Low}(Z)$ contains the lower bounds of real numbers in Z, and $\text{Inf}(Z)$, the **infimum** of Z, is the greatest lower bound of all real numbers in Z if one such bound exists.

An **integer** is a natural number, the number 0, or a number of the form $-n$, where n is a natural number. A **rational number** is a number that can be written in the form p/q with an integer p and a natural number q. A real number that is not rational is called **irrational**. A **complex number** is the same as a **number** (though sometimes the term is used to express that a number is not real).

Note that in the traditional constructive approach to complex numbers there are several distinct natural constructions of $\mathbb{C}$ from the real numbers, among them

- $\mathbb{C}$ is the subalgebra of real 2×2-matrices A with $A_{11} = A_{22}$ and $A_{12} = -A_{21}$;
- $\mathbb{C}$ is the quotient field of real polynomials in an indeterminate x modulo the principal ideal generated by $x^2 + 1$.

The two constructions have different accidental properties not inherent in the concept of complex numbers as actually used. (Worse things can happen: In the construction of the complex numbers I learnt as a student, the pair $(1, 2)$ was at the same time a representative for the number $1/2$ and for the number $1 + 2i$.) Hence neither can be regarded as defining *the* complex numbers. The latter are generally used as uniquely determined objects characterized by their familiar arithmetic and topological properties and nothing else – rather than by a particular construction.

The FMathL axioms A11–A13 select some of these properties, sufficient for deriving all other construction-independent properties of complex numbers. Indeed, Neumaier [53] shows that a variant of Axioms A11–A13 (easily seen to be equivalent to these) implies all usual properties of complex, real, rational, integral, and natural numbers. Defining the algebraic structures $\mathbb{Z}$, $\mathbb{Q}$, and $\mathbb{R}$ is possible within FMathL but requires tools that go beyond this foundational document in that we need to reflect the concept of a structure satisfying given axioms.

On the other hand, the above constructions (and several others) are useful as models of fields isomorphic to $\mathbb{C}$ and can of course be used within FMathL in the same way as different models of the finite simple group of order 60 can be used to illuminate its structure.

2.9 Texts

Texts, formed by concatenating symbols called characters, are the dominant communication medium of mathematics. Mathematicians do not agree on the collection of characters to be used; they feel entitled to invent new characters as they need them. Thus in the text reflection axioms, FMathL only specifies that the decimal digits are among the characters.

Axiom A14.
We say that an object x is a **character** if $x \in \text{Char}$, and a **text** if $x \in \text{Text}$. Digits are characters. Every character is a text. The **empty text** Null is a text; it is not a character. If x and y are texts then $x \,\&\, y$ is a text, the **concatenation** of x and y.

In particular, every text (and hence every character) exists.

Although our setting is intrinsically typeless, we see here that certain named objects such as Char or Text may function as a kind of simple types if they contain precisely the intended objects of a certain type. However, unlike in type theories, the same object may belong to several "types" in the present sense, without "types" being explicitly dependent upon each other.

Note that the metacharacter & is a primitive operation and hence by Axiom A3 *not* an object, hence not a character! The same holds for the metacharacters $\wedge$, $=$, $\in$, $+$, etc.. This means that when reflecting (in a later paper) metatexts on the object level, there must be some form of encoding of these metacharacters, similar to the need for encoding a quotation mark in the usual quoting mechanism of text inside programming languages.

Axiom A15.
We require for all texts x, y and characters c, d that
(P18) $x \,\&\, y = \text{Null}$ iff $x = \text{Null}$ and $y = \text{Null}$;
(P19) $x \,\&\, c = y \,\&\, d$ implies $x = y$ and $c = d$;

(P20) If Z is an object with $\textsc{Null} \in Z$ such that $x \,\&\, c \in Z$ for every text $x \in Z$ and every character c, then $z \in Z$ for every text z.

(P20) says that all texts are obtained from the empty text by appending characters, and is a text version of the induction principle. The **cancellation laws**

$$\begin{aligned} x \,\&\, y = x \,\&\, z &\quad\Rightarrow\quad y = z, \\ x \,\&\, z = y \,\&\, z &\quad\Rightarrow\quad x = y \end{aligned}$$

can be derived from A15.

2.10 Application

In FMathL, like in combinatory logic, any object may be applied to any other object. We recall our convention from Section 2.2 to write

$$\begin{array}{lll} f(x) & \text{for} \quad f \,@\, x, & \quad (\textbf{application}) \\ f(x,y) & \text{for} \quad f(x)(y), & \quad (\textbf{currying}) \end{array}$$

with a similar interpretation for $f(x, y, z)$, etc.. We read $f(x)$ as f **of** x or f **at** x, and call it the **value** of f at x. We read $f(x, y)$ as f **of** x and y, etc..

Axiom A16.
We say that the object f is **defined on** the existing object Z if $x \in Z$ implies that $f(x)$ exists. If the object f is defined on the object Z then the **image**

$$f[Z] := f \square Z$$

of Z under f exists, and

(P21) $z \in f[Z]$ iff $f(x) = z$ for some $x \in Z$.

Axiom A17.
A **predicate** is an object f such that $f(x) = \mathbf{0}$ or $f(x) = \mathbf{1}$ whenever $f(x)$ exists. If the predicate f is defined on the object Z then the **selection** $Z \,|\, f$ of Z through f exists, and

(P22) $x \in Z \mid f$ iff $x \in Z$ and $f(x)$.

The attempted assertion of a predicate f at a particular argument x is 3-valued, with values true if $f(x) = \mathbf{1}$, false if $f(x) = \mathbf{0}$, and nominal otherwise.

(P21) and (P22) entitle us to write as customary

$$\begin{array}{lll} \{f(x) \mid x \in Z\} & \text{for} & f[Z], \\ \{x \in Z \mid f(x)\} & \text{for} & Z \mid f, \\ \{f(x) \mid x \in Z, g(x)\} & \text{for} & f[\{x \in Z \mid g(x)\}]. \end{array}$$

Note that $\{f(x) \mid x \in Z\}$, $\{x \in Z \mid f(x)\}$, or $\{f(x) \mid x \in Z, g(x)\}$ may exist even when $f(x)$ or $g(x)$ is nominal for some $x \in Z$, since x is a bound variable with only syntactic meaning that can be eliminated by undoing the above abbreviations. This is the only nontrivial way nominal objects can meaningfully appear in asserted expressions.

2.11 Functions

We now reflect functions, which encode mathematical transformations, and show how expressions can be used to define such functions. This is done by means of the object FUN whose application to an existing object f transforms it into a function FUN(f).

Among objects, functions are distinguished by an extensionality property.

Axiom A18.
We say that the object f is a **function** if FUN$(f) = f$. For every existing object f, the object FUN(f) is a function such that, for all objects x,
(P23) FUN$(f)(x) = f(x)$ if $f(x)$ exists, FUN$(f)(x)$ is nominal otherwise.
If f and g are functions then
(P24) FUN(f) = FUN(g) iff both $f(x) = g(x)$ for all objects x for which $f(x)$ and $g(x)$ exist, and $f(x)$ is nominal exactly when $g(x)$ is nominal.

The assertion of FUN$(f) = f$ implies that f exists, hence every function exists. (P24) expresses the so-called **extensionality** of functions.

We use the term function with the meaning familiar from the λ-calculus. In the traditional mathematical language, the term function may, dependent on the context, have a different meaning. Indeed, as a point is just a member of a space, and a vector is just a member of some vector space, so a function is just a member of a function space. For example, the elements of the space of square integrable functions on the reals are equivalence classes of certain maps, and the elements of the space of rational functions of x with coefficients in some field are formal rational expressions in x. In both cases, we do not have functions in the present sense. We regard this as part of the context-dependent ambiguity of the common mathematical language.

Axiom A19.
For arbitrary existing objects x, y, f, and g, and arbitrary intrinsic operations ω, the objects CHOICE, ID (the **identity**), CONST(y), and POINTWISE$(f\,\omega\,g)$ are functions satisfying, for all existing objects x and y,
(P25) $x \in Z$ implies CHOICE$(Z) \in Z$.
(P26) CHOICE(Z) is nominal if Z contains no object.

(P25), the **axiom of global choice**, allows us to choose a distinguished element from an object, provided that there is such an element: $x :=$ CHOICE(A) is the formal version of "Let x be some distinguished element from A"; the element $x \in A$ is distinguished by writing this. In particular, CHOICE$(\{x\}) = x$ if x exists, and CHOICE$(\{x\})$ is nominal otherwise. Note that for existing x and y it is undecided whether CHOICE$(\{x, y\})$ is x or y – different implementations of the mathematical framework might make different distinguished choices, if they decide the problem at all.

Axiom A20.

(P27) $\text{I}_{\text{D}}(x) = x$;

(P28) $\textsc{Const}(y)(x) = y$;

(P29) $\textsc{Pointwise}(f\,\omega\,g)(x) = f(x)\,\omega\,g(x)$ if $f(x)\,\omega\,g(x)$ exists,
$\textsc{Pointwise}(f\,\omega\,g)(x)$ is nominal otherwise.

Note that for nominal objects x, the object $\textsc{Id}(x)$ is nominal and does not satisfy $\textsc{Id}(x) = x$, and $\textsc{Const}(x)(y)$ is nominal and does not satisfy $\textsc{Const}(x)(y) = x$. Thus $\textsc{Id}$ is the identity function on existent objects (only), and $\textsc{Const}(x)$ the function with constant value x for existing arguments (only).

This axiom allows us (among other uses) to define the **composition** $\circ$ (**after**) of objects by writing

$$f \circ g \quad \text{for} \quad \textsc{Pointwise}(\textsc{Const}(f)(g)).$$

Indeed, the conventional composition formula

$$(f \circ g)(x) = f(g(x))$$

follows from

$$\begin{aligned}(f \circ g)(x) &= \textsc{Pointwise}(\textsc{Const}(f)(g))(x)\\ &= \textsc{Pointwise}(\textsc{Const}(f)\,@\,g)(x)\\ &= \textsc{Const}(f)(x)\,@\,g(x) = f\,@\,g(x) = f(g(x)).\end{aligned}$$

2.12 Abstraction

Abstraction – the creation of a function from an expression – is probably the most difficult, but also the most powerful basic principle in mathematics. It is indispensable for reflecting the interpretation of expressions as functions in which we can substitute variables by objects, and accounts to a large extent for the flexibility of the mathematical language.

$\textsc{Id}$, $\textsc{Const}$, and $\textsc{Pointwise}$ are similar to the combinators I, K, and S in combinatory logic, and play an analogous role. As in the λ-calculus, they allow us to give a precise formal semantics for what it means to define a function by an expression.

We write $E\Big|_{z=x}$ for the expression obtained from E by replacing each occurrence of the variable z by x.

2.12.1 Theorem. *For every expression E and every variable z, there is a unique, constructively defined function f such that $f(x) = E(x)$ for every object x for which $E(x)$ exists and $f(x)$ is nominal otherwise.*

Here $E(x)$ is the conventional abbreviation for $E\Big|_{z=x}()$; the dependence on the choice of the variable z is suppressed.

Proof. That there is at most one such function follows from the extensionality of functions. Therefore we only need to show existence.

For an expression of length 1, the expression E is a name. If this name is z then $f := \textsc{Id}$ satisfies $f(x) = \textsc{Id}(x) = x = E(x)$; otherwise, $f = \textsc{Const}(E())$ satisfies $f(x) = \textsc{Const}(E())(x) = E() = E(x)$. Thus the assertion holds for all expressions of length 1. Now suppose that the assertion holds for all expressions of length $< L$. If E is an expression of length $L > 1$ then it has the form $(E' \,\omega\, E'')$ with subexpressions E' and E'' of length $< L$, and we have uniquely defined functions f' and f'' satisfying $f'(x) = E'(x)$ and $f''(x) = E''(x)$. Therefore, the function $f := \textsc{Pointwise}(f' \,\omega\, f'')$ satisfies $f(x) = \textsc{Pointwise}(f' \,\omega\, f'')(x) = f'(x) \,\omega\, f''(x) = E'(x) \,\omega\, E''(x) = E(x)$. Thus $f(x) = E(x)$. By induction, the assertion holds for expressions of arbitrary length. $\square$

A standard substitution shows that the theorem also extends to expressions involving defined operations, as long as all defined operations that occur are defined in terms of an expression, without reference to words or a metaconcept. Using currying, everything easily extends to definitions of functions f of the form $f(x, y) := E(x, y)$, etc..

Definition by abstraction. In the λ-calculus, the function f from Theorem 2.12.1 is denoted by

$$f := \lambda z.E$$

Mathematicians usually do not use the λ-notation outside of λ-calculus. Instead, they refer to the function f from Theorem 2.12.1 as the function defined by

$$f(x) := E(x)$$

for every object x for which $E(x)$ is defined, and write

$$\begin{array}{lll}
\{E(x) \mid x \in Z\} & \text{for} & (\lambda z.E)[Z], \\
\{x \in Z \mid E(x)\} & \text{for} & Z \mid (\lambda z.E), \\
\{x \in Z \mid A(x), B(x)\} & \text{for} & \{x \in Z \mid A(x) \wedge B(x)\},
\end{array}$$

etc., thus eliminating the need of explicit abstraction with the λ-notation. Using these conventions significantly enhances readability. For example, the left hand side of

$$\{x \in X \mid x \in Y\} = X \mid (\lambda x.(x \in Y)) = \textsc{Pointwise}(\textsc{Id} \in \textsc{Const}(Y))$$

is much more expressive than the term involving the λ-expression or the explicit construction involving the combinators, as it immediately reveals the meaning of the expression as the intersection of X and Y. Essentially the same argument allows us to reflect complementation. For arbitrary objects X and Y, we write

$$X \setminus Y \quad \text{for} \quad \{x \in X \mid x \notin Y\}, \qquad (\textbf{complement})$$

Clearly,

$x \in X \setminus Y$ iff $x \in X$ and $x \notin Y$.

The explicit representation of the function f from Theorem 2.12.1 by an explicit abbreviating definition quickly gets incomprehensible. For example, if f is an object, the object $g := \lambda x.(f(x) = 0 \Rightarrow x = 0)$ can be constructed as

$$g := \textsc{Pointwise}\Big(\textsc{Pointwise}(f = \textsc{Const}(0)) \Rightarrow \textsc{Pointwise}(\textsc{Id} = \textsc{Const}(0))\Big).$$

Indeed,

$$\begin{aligned} g(x) &= \textsc{Pointwise}\Big(\textsc{Pointwise}(f = \textsc{Const}(0)) \Rightarrow \textsc{Pointwise}(\textsc{Id} = \textsc{Const}(0))\Big)(x) \\ &= \Big(\textsc{Pointwise}(f = \textsc{Const}(0))(x) \Rightarrow \textsc{Pointwise}(\textsc{Id} = \textsc{Const}(0))(x)\Big) \\ &= \Big(f(x) = \textsc{Const}(0)(x) \Rightarrow \textsc{Id}(x) = \textsc{Const}(0)(x)\Big) \\ &= (f(x) = 0 \Rightarrow x = 0). \end{aligned}$$

In practice, one therefore never creates the explicit formula exhibiting the fully expanded definition of an object defined by means of constructions justified by defined abbreviations and proved existence theorems. This is analogous to the practice that one virtually never writes out natural numbers such as 8 or 10^9 in terms of their nearly incomprehensible explicit construction with a successor operation.

We now discuss further results that show the power of abstraction. Abstraction enables us to prove that no object contains all other objects:

2.12.2 Theorem. *For every object Z, there is an object z with $z \notin Z$.*

Proof. The proof is the well-known argument by Russell [65] that, applied to naive set theory, leads to contradiction. By Axiom A16, the object $z := \{x \in Z \mid x \notin x\}$ exists. By (P22), $x \in z$ iff $x \in Z$ and $x \notin x$. Applied to $x = z$, this shows that $z \in z$ iff $z \in Z$ and $z \notin z$. Now either $z \in z$ or $z \notin z$, and in both cases we conclude that $z \in Z$ is false. Hence $z \notin Z$. □

Abstraction also allows us to define the **restriction** $f\Big|_Z$ of an object f to an object Z by

$$f\Big|_Z(x) := \textsc{Choice}(\{y \in f[Z] \mid x \in Z \wedge y = f(x)\}),$$

The restriction to Z of the function $f = \lambda z.E$ is generally referred to as the function g defined (on Z) by

$$g(x) := E(x) \quad \text{for } x \in Z$$

whenever $E(x)$ is defined.

Abstraction also allows us to define various forms of functions defined by case distinctions. For example, if Y contains all $a(x)$ and $b(x)$ with $x \in Z$ then

$$f(x) := \textsc{Choice}\Big(\Big\{y \in Y \;\Big|\; (y = a(x) \wedge c(x) = 1) \vee (y = b(x) \wedge c(x) = 2)\Big\}\Big)\Big|_Z$$

defines the function with

$$f(x) = \begin{cases} a(x) & \text{if } c(x) = 1, \\ b(x) & \text{if } c(x) = 2, \end{cases}$$

and $f(x)$ is nominal otherwise.

2.13 Categories

The concept of a mathematical structure is captured in the notion of a category. Informally, a category is a transitive directed graph whose vertices (certain objects) may be connected by arbitrarily many directed edges (called arrows). Transitivity says that two arrows can be composed to a third arrow (their product) if the tail (domain) of the first arrow agrees with the tip (codomain) of the second arrow. In many (but not all) categories, the vertices are structured sets and the arrows are structure-preserving mappings from their domain to their codomain; cf. Axiom A25 below.

We write

$$\begin{array}{lll} \mathrm{ID}_A & \text{for} & \mathrm{ID}\_(A); \\ f : A \to B & \text{for} & f \in (A \to B), \\ f, g : A \to B & \text{for} & f : A \to B \wedge g : A \to B, \end{array}$$

with a similar interpretation for $f, g, h : A \to B$, etc..

Axiom A21.
An object f is called an **arrow** from $\mathrm{DOM}(f)$, the **domain** of f, to $\mathrm{COD}(f)$, the **codomain** of f, if $\mathrm{DOM}(f)$ and $\mathrm{COD}(f)$ exist. The **arrow product** $g \diamond f$ of two arrows g and f exists iff $\mathrm{DOM}(g) = \mathrm{COD}(f)$ and is then an arrow from $\mathrm{DOM}(f)$ to $\mathrm{COD}(g)$. For any object A, the **identity** ID_A on A exists. For arrows f, g, h and objects A, B, C, D,

(P30) $f : A \to B$ iff $\mathrm{DOM}(f) = A$ and $\mathrm{COD}(f) = B$;
(P31) if $f : A \to B$, $g : B \to C$, $h : C \to D$ then $h \diamond (g \diamond f) = (h \diamond g) \diamond f$;
(P32) $\mathrm{ID}_A : A \to A$.
(P33) $f : A \to B$ implies $\mathrm{ID}_B \diamond f = f$ and $f \diamond \mathrm{ID}_A = f$.

It is customary in mathematics to talk about the arrow $f : A \to B$, meaning the arrow f with $f : A \to B$.

Axiom A22.
An object C is called a **category** if $\mathrm{HOM}(C)$ exists. The elements of $\mathrm{HOM}(C)$ are called **homomorphisms** (or C-**morphisms**) of the category C. An object A is called a **structure** of the category C if $C(A) = A$. For every structure A of the category C, the object ID_A is a homomorphism of C. For every homomorphism f of the category C, the objects $\mathrm{DOM}(f)$ and $\mathrm{COD}(f)$ are structures of C. For any category C, any object A, and any two arrows f, g,

(P34) $A \in C$ iff $C(A) = A$;
(P35) $C(A) \in C$ iff $C(A)$ exists;
(P36) $f, g \in \mathrm{HOM}(C)$ implies $g \diamond f \in \mathrm{HOM}(C)$ if $g \diamond f$ exists.

Note that, in FMATHL, a category contains all its objects. This is possible without difficulties since $\in$ is not extensive. Thus FMATHL does not need a separate *Ob* operator.

If $C(A)$ exists, it is the structure of category C associated with the object A. The nature of this association depends on the category and must be postulated in each case; see, e.g., the case of the category $\textsc{Set}$ introduced in Section 2.14 below. In many cases, $A \to C(A)$ is an appropriate forgetful functor.

Many categories are given the characteristic name conventionally attached to their structures. For example, the structures of the category $\textsc{Set}$ are called sets, the structures of the category $\textsc{Group}$ (not introduced here) are called groups, etc..

Axiom A23.
We say that the object Z is **small** if $Z \in \textsc{Small}$. If Z is small and $Y \subseteq Z$ then Y is small. If Z is small and f is defined on Z then the image $f[Z]$ is small. If Z is small and C is a category then $C(Z)$ is small if it exists.

In particular, every small object exists.

2.14 Sets

The concept of a set is reflected by the following axiom.

Axiom A24.
An object A is called a **set** if $\textsc{Set}(A) = A$. We say that A is a **subset** of the object Z if A is a set and $A \subseteq Z$. The object $\textsc{Set}$ is a category satisfying

(P37) $\textsc{Set}(A) = (A \mid \textsc{Const}(\mathbf{1}))$;

(P38) If A and B are sets then $A = B$ iff $x \in A \Leftrightarrow x \in B$.

The **empty set** $\emptyset$ is a set. If the objects x and y exist then $\{x, y\}$ is a set. If A is a set and every $B \in A$ is a set then $\textsc{Union}(A)$ is a set. If the object f is defined on the set A then the image $f[A]$ of A under f and the set restriction $A \mid f$ of A by f are sets.

Every set exists since the assertion of $\textsc{Set}(A) = A$ implies the existence of A. (P38) expresses the basic property of sets, their **extensionality**.

If A is a set, there is a set B with $B \not\in A$. This is proved exactly as in Theorem 2.12.2. Thus no set contains all sets.

Axiom A25.
An object $f \in \textsc{Hom}(\textsc{Set})$ is called a **map** (or **mapping**); it is called a **map from** the set A to the object Z if $f : A \to Z$. In this case, $f[A]$ is called the **range** of the map f. For any two maps $f, g : A \to Z$,

(P39) $f = g$ iff $f(x) = g(x)$ for all $x \in A$;

(P40) $f(x)$ exists iff $x \in A$;

(P41) $x \in A$ implies $f(x) \in Z$;

(P42) $h \diamond f = h \circ f$ if Z is a set and $h : Z \to Y$;

(P43) $\textsc{Id}_A = \textsc{Id}\Big|_A$.

(P39) is the extensionality of maps.

A map $f : A \to Z$ is called **injective** if $f(x) = f(y)$ implies $x = y$, **surjective** if $f[A] = \text{Cod}(f)$, and **bijective** if it is injective and surjective.

Note that the notions of a map and a function differ in FMathL: Unlike a map, a function does not have the concepts of domain or codomain associated with it. In particular, the notions of surjectivity and bijectivity make sense only for maps and not for functions. Since $f = g$ implies $\text{Cod}(f) = \text{Cod}(g)$, two maps $f : A \to Z$ and $g : A \to Y$ are different if $Z \neq Y$ even when $f(x) = g(x)$ for all $x \in A$, while the associated functions $f\Big|_A$ and $g\Big|_A$ are equal. To go from a function to a corresponding map, we need an additional axiom:

Axiom A26.
If f is defined on the set A and $f[A] \subseteq Z$ then $g := \text{Map}(A, Z, f)$ is the arrow $g : A \to Z$ with $g(x) = f(x)$ for all $x \in A$.

(Remember that $\text{Map}(A, Z, f) = \text{Map}(A)(Z)(f)$ by currying.)

Nevertheless, consistent with current practice, we loosely identify g with f on the informal language level, and thus allow the use of $f : A \to Z$ to also (ambiguously) refer to a function f defined on A with values in Z.

Tuples and Cartesian products. If m and n are natural numbers, we write

$\{m : n\}$	for	$\{x \in \mathbb{N} \mid m \leq x \leq n\}$,	(**range of natural numbers**)
$k = m, \ldots, n$	for	$k \in \{m : n\}$.	

If n is a natural number or zero, an n-**tuple** is a function t for which $t(x)$ exists iff $x \in \{1 : n\}$. For $k = 1, \ldots, n$, one writes t_k in place of $t(k)$, and calls t_k the k-th (or first if $k = 1$, second if $k = 2$, third if $k = 3$) **entry** of t. In terms of the entries, one writes $t = (t_1, \ldots, t_n)$. A 2-tuple t is called a **pair**, and one writes $t = (t_1, t_2)$. A 3-tuple t is called a **triple**, and one writes $t = (t_1, t_2, t_3)$; etc.. For any existing object Z, the domain

$$Z^{\times k} := (\{1 : k\} \to Z)$$

consists of all k-tuples with entries in Z; it is a set if Z is a set. An n-**ary relation** on an existing object Z is an object R with $R \subseteq Z$ One writes **nullary**, **unary**, **binary**, and **ternary** for n-ary when $n = 0, 1, 2, 3$, respectively.

For arbitrary existing objects X and Y, we define their **Cartesian product**

$$X \times Y := \{(x, y) \in (X \cup Y)^{\times 2} \mid x \in X, y \in Y\};$$

in particular, $Z^{\times 2} = Z \times Z$ if Z exists. Similarly, we define

$$X \times Y \times Z := \{(x, y, z) \in (X \cup Y \cup Z)^{\times 3} \mid x \in X, y \in Y, z \in Z\},$$

etc., so that $Z^{\times 3} = Z \times Z \times Z$, etc.. Note that $X \times Y \times Z$, $(X \times Y) \times Z$, and $X \times (Y \times Z)$ are different. We write

$$\begin{array}{lll} \{f(x,y) \mid x,y \in Z\} & \text{for} & \{f(t_1,t_2) \mid t \in Z^{\times 2}\}, \\ \{f(x,y) \mid x \in X, y \in Y\} & \text{for} & \{f(t_1,t_2) \mid t \in X \times Y\}, \end{array}$$

and similarly for related expressions.

Axiom A27.
If Z is a small object and A is a small set then $A \to Z$ is a set. If Z is a small object then $\textsc{Subsets}(Z)$ is a set satisfying
(P44) $B \in \textsc{Subsets}(Z)$ iff $B \subseteq Z$;
(P45) $\textsc{Union}(\textsc{Subsets}(Z)) = \textsc{Set}(Z)$;
(P46) $\textsc{Subsets}(\textsc{Text})$ is small.

Thus $\textsc{Set}(Z)$ is the set consisting of all elements of the small object Z.

The reason why we require $\textsc{Subsets}(\textsc{Text})$ to be small is that this enables us to reflect all arguments about context logic given in Neumaier & Marginean [56], and thus constitutes a minimal requirement to reflect and reason about the FMathL foundations in FMathL itself. Note that $\mathbb{C}$ can be proved to be part of an image of $\textsc{Subsets}(\textsc{Text})$, hence is small, too.

Chapter 3

Completing the foundations

3.1 Mathematical frameworks

With Axiom A27, our axiom system is complete. Any mathematical structure satisfying these axioms will be called a **mathematical framework**. Informally, a mathematical framework $\mathbb{O}$ may be viewed as a kind of metacategory of all categories. Indeed, the objects and arrows of $\mathbb{O}$ form a category on the subject level, with considerable additional structure.

It turns out, and will be shown in a subsequent paper (NEUMAIER & SCHODL [57], that one can reflect the full machinery used to define mathematical frameworks inside each particular mathematical framework; cf. Section 3.4 below. In each subject level where a mathematical framework is given, the FMATHL mathematical framework is the particular, distinguished implementation of a mathematical framework constructed inside the given mathematical framework according to the description given in [57].

Negation laws derived. We could have defined an intuitionistic mathematical framework in which Axiom A6 is replaced by $\neg\mathbf{0} = \mathbf{1}$ and a disjunction satisfying **or reflection**, i.e., $x \vee y \in \overline{\Gamma}$ iff $x \in \overline{\Gamma}$ or $y \in \overline{\Gamma}$, instead of being defined in terms of $\wedge$.

However, a slight modification of an argument of GOODMAN & MYHILL [32] shows that this already implies the negation laws of Axiom A6. Indeed, $\mathbf{0} = \neg\mathbf{1}$ follows directly from the reflexivity of equality.

To prove the double negation law $\neg\neg p = p$ for statements p, using only intuitionistically valid arguments, let p be a statement satisfying $\neg\neg p$. To show that p holds, we define two maps $f, g : \{\mathbf{0}, \mathbf{1}\} \to \{\mathbf{0}, \mathbf{1}\}$ by

$$f(x) := p \vee \neg x, \quad g(x) := p \vee x \quad \text{for } x \in \{\mathbf{0}, \mathbf{1}\},$$

and consider the function c defined by

$$c(f) := \text{CHOICE}(\{x \in \{\mathbf{0}, \mathbf{1}\} \mid f(x)\}).$$

Since $f(\mathbf{0})$ holds, $f(c(f))$ holds, hence $p \vee \neg c(f)$ by definition of f. Similarly, since $g(\mathbf{1})$ holds, $g(c(g))$ holds, hence $p \vee c(g)$ by definition of g. Combining both conclusions, we get $p \vee (\neg c(f) \wedge c(g))$. Thus p (and hence Axiom A6) holds if we can show that

$$z := \neg c(f) \wedge c(g)$$

is false. We show this by contradiction, and therefore assume that z holds. Then $c(f) = \mathbf{0}$ and $c(g) = \mathbf{1}$, hence $f = g$ is false. But if p holds then $f(x) = \mathbf{1} = g(x)$ for $x = \mathbf{0}$ and $x = \mathbf{1}$, hence $f = g$ by extensivity of maps, contradiction. Thus p is false. Thus our assumption that z holds implies $\neg p$. But $\neg\neg p$ implies that $\neg p = \mathbf{0}$. Hence z implies $\mathbf{0}$, contradiction.

3.2 Foundations for mathematics

Mathematics may be considered as the science and art of precise concepts and their relations. It plays a fundamental role in many aspects of modern human life, because precise concepts are needed and usefully employed to predict and control the complexities of our present culture.

While current automatic theorem provers fail upon the slightest slip of the pen, mathematics as it is practiced today is very robust. It is nearly independent of changes in foundations, conflicts in details, sloppiness of exposition, small errors and inaccuracies, etc..

We learn even from slightly incorrect theories. Most standard works contain mistakes, but are nevertheless extremely useful and influential. When the intuition behind a mathematical text is correct, the correct interpretation of the text can usually be inferred, so that readers can reconstruct a correct version if they are sufficiently interested.

This calls for an explanation. In my opinion, mathematics is intelligible and trustworthy because of its everywhere locally verifiable context, which is due to its organization as an extensive deduction graph that tolerates weak links and allows one to identify these. Indeed, often mathematicians only read key statements and try to prove them for themselves without recourse to the original proof – the latter only serves as a backup in case the own proof attempts failed.

The modular nature of mathematics makes it robust even against consistency problems in the foundations. The fact that one can work fruitfully in inconsistent systems and get meaningful mathematics distinguishes mathematics from logic. It was exemplified by the development and usefulness of naive set theory in the period 1874–1899, before its inconsistency was discovered. The reason is that theory and proofs connect concepts rather than build up a unique logical building. Also, while working towards a proof by contradiction, one always works in an inconsistent system!

Thus, even if somewhere something fails, it will not affect everything but only a small part of the whole edifice. Therefore, not the *logical* proof – anyway available only for a tiny fraction of today's mathematical edifice – but the *mathematical* proof is the essence of the reliability of mathematics in the real world.

By Gödel's second incompleteness theorem, since mathematics contains Peano arithmetic, we cannot even in principle know of any mathematical definition whether it makes sense in the sense of not producing a contradiction: Even definitions leading to conservative extensions of a theory need to assume on trust that the theory one starts with is consistent in the first place. We only can know when something fails to make sense.

One therefore generally simply assumes that definitions are consistent. If a contradiction is found, one inspects the deduction graph to identify a likely weak link that should not have been trusted. Also, typically, one tries to isolate the contradiction into a simple paradox, so that one understands it and can guard the remaining theory from falling prey to the same paradox. After some repair work that usually succeeds, the mathematical edifice stands again unthreatened.

Communicating mathematics. The communication of mathematics between two different subjects poses a problem since these generally have implemented the object level in different ways. This is obviously the case in human-machine communication, where the fundamental differences between human brains and silicon computers must be bridged.

It is also the case with human thinking, where each brain constitutes an operating system with slightly different hardware. The implementation of mathematics in a human brain is achieved through an education process that may produce in different people quite different implementations of the concepts, and hence lead to quite different, subjective elements in the intuition about these concepts and their relations. (Are your real numbers infinitely long decimal numbers? nested sequences of intervals? equivalence classes of Cauchy sequences? Dedekind cuts? particularly nice surreal numbers? or something else?)

How is it then possible that subjects (whether humans or machines) with very different subject levels can communicate objectively? As we know, it works to a considerable extent in ordinary life, though imperfect communication often gives rise to misunderstandings. It works much better in mathematics, due to its highly structured approach, optimized for clear communicability.

It seems to me that communication works in science (which includes mathematics as the science of precise concepts and their relations) since scientific statements are heavily constrained. The assumption that all scientific statements made by subject X are goal-directed and meaningful for subject X in their context, with a meaning closely reproducible in the subject level of any educated receiving subject Y, is a severe constraint on texts serving as such statements. We all notice occasional misunderstandings if our communication partner responds to one of our statements in a way that does not make sense, and good communication skills include the ability to notice such misunderstandings and to have protocols for exposing, discussing, and overcoming them. **Understanding** – i.e., a common internal representation of the objects of discourse in both partners of what has been communicated – is achieved when the subsequent communication shows that no further misunderstandings occur.

In mathematics, where all communication is based on a finite formal basis, such an understanding can be achieved in finitely many steps.

In particular, communication is easy if the communication subjects agree on a common mathematical framework, so that both subject levels agree on the object level as far as necessary, by satisfying the specifications. Learning this common mathematical framework is a finite task for humans, and programming a computer to understand this mathematical framework is a finite task, too.

Therefore, the object level defined by the FMathL axioms for a mathematical framework may indeed be considered to be objective, in the sense that it makes unambiguous communication possible.

Foundations of mathematics. The foundations of mathematics may similarly be regarded as the science and philosophy of the natural laws by which mathematicians define and reason successfully. Thus we may suppose that there is some underlying substance of precisely communicable concepts to be modelled by FMathL.

FMathL tries to define a language sufficient to express all objectively communicable mathematical concepts. In the FMathL sense, **objectively communicable concepts** are objects that are provably existent in the mathematical framework of nameable objects discussed in Section 3.5. They exist in many possible (subjective) implementations, but their essence – given by the way they are embedded in this framework – is unique up to isomorphism, and hence objective.

This essence may be regarded by those with a disposition towards a Platonic view of mathematics as constituting the **Platonic world** of **ideas**, i.e., of objectively communicable concepts. Indeed, a number of Platonic dialogues, e.g., Hippias major and Hippias minor, can easily be viewed as teaching precision in the communication of abstract concepts, with Socrates as the teacher who, according to his victim Hippias, dissects the language into *"scrapings and shavings of discourse, divided into bits"* (Plato [63]) in order to find out what precisely is agreed upon, and occasionally gets into deadlocks. – Hippias: *"I cannot agree with you, Socrates, in that."* Socrates: *"Nor I with myself, Hippias; but that appears at the moment to be the inevitable result of our argument"* (Plato [64]). – The deadlocks are due to a confusion of different concepts with the same name. The ambiguity of language has been a difficulty from the start....

The FMathL setting of the Platonic world is thus slightly different from that discussed by Penrose [61]: He thinks of the physical world, the world of Platonic forms, and the world of the human mind as being three separate worlds. In FMathL, the world of the human (or electronic) mind is modelled as a multiplicity of subject levels, while the world of Platonic forms is what is common to the object levels represented inside these subject levels.

As in physics, our view of this underlying reality may turn out to be inadequate as new evidence is discovered – in mathematics in the form of contradictions in present axiom systems, or of new, more powerful concepts. The old views must then be repaired by modifying them to account for the new evidence.

As physicists know that the deepest layers of Nature always remain unknown, so mathematicians know (and can even prove) that the deepest layers of the realm of mathematical thought (or Platonic reality) always remain obscure. However, the belief that reality is consistent (after all, it exists and works!) drives the belief in the existence of a repaired view whenever the currently accepted view is found inadequate. This replaces the lack of consistency proofs that, by Gödel's second incompleteness theorem, is unavoidable for foundations in which one can do all standard mathematics.

In the words of BOURBAKI [13], *"we believe that mathematics is destined to survive, and that the essential parts of this majestic edifice will never collapse as a result of the sudden appearance of a contradiction; but we cannot pretend that this opinion rests on anything more than experience."*

Thus foundations of mathematics are not like the foundations of a building, which crashes if its foundations are found defective, but rather like the roots of a tree, which grow and get improved and refined as the tree grows.

As in all natural sciences, the adequacy of the FMathL mathematical framework as a foundation of mathematics, i.e., its consistency and sufficient generality, is given an empirical justification only.

The consistency of the FMathL mathematical framework has not been investigated so far, apart from checking that the traditional paradoxes do not cause harm. At present, I am also not completely sure (though confident) that all standard mathematics can be developed in the FMathL mathematical framework.

Future versions of FMathL will have outgrown deficiencies that the current version still might have.

Strengthening the FMathL mathematical framework by requiring additional distinguished objects and/or additional assumptions (e.g., that every set is small) is of course possible in the same way as, for example, in set theory, the existence of inaccessible cardinals may be assumed in addition to ZFC.

3.3 Infinity

An object Z is called **infinite** if no object f exists such that

$$Z \subseteq f[\{x \in \mathbb{N} \mid x \leq n\}]$$

for some number n. For example, $\mathbb{C}$, $\mathbb{N}$, and TEXT are infinite.

On the other hand, FMathL regards all objects – among them the distinguished objects, statements, numbers, texts, functions, and sets – as finite (i.e., finitely describable) things on the subject level. The finite description of objects is accomplished through (abbreviations for) finite expressions in a few "free" variables, as follows.

The metaset of the strings denoting expressions for the elements of the free mathematical framework in these variables is countable and becomes a mathematical framework $\mathbb{O}_{\text{name}}$ – a **framework of nameable objects** – by declaring intrinsic operations on these strings in the natural way, and by interpreting the assertion of a string as saying that, in the free mathematical framework, the expression corresponding to the string is equal to **1**, the true statement. While this cannot always be decided constructively, any proved assertion can be checked for correctness.

It is clear that all mathematics that can be expressed in some mathematical framework can be expressed in a framework of nameable objects. This holds even when – relaxing Axiom A1 – our original mathematical framework $\mathbb{O}$ of objects were an uncountable metaset: One would obtain the countable mathematical framework $\mathbb{O}_{\text{name}}$ of nameable objects in which everything expressible in $\mathbb{O}$ can be equivalently expressed. This is the FMathL version of the Löwenheim-Skolem theorem, and justifies a posteriori the countability assumption in Axiom A1.

The unnameable part of the metaset of objects is irrelevant (i.e., subjective, implementation-dependent) for mathematics since one cannot refer to it objectively. Reflecting Wittgenstein [72], who said, "Wovon man nicht sprechen kann, darüber muß man schweigen" – of what one cannot talk about, one must be silent –, FMathL makes this observation part of the foundations.

Thus, in a mathematical framework of nameable objects, all objects are metafinite, although some of them are infinite. (When object level and subject level are not distinguished, this can be seen as a variant of Skolem's paradox discussed in Section 3.5.) This makes standard mathematics as constructive as anything dealing with the infinite can be.

In such a constructively obtained mathematical framework of nameable objects, membership is a form of "mental" association rather than "physical" containment. Note that the mind referred to in "mental" may be that of a human or that of a machine. This allows for example the finite distinguished object $\mathbb{C}$, the domain of complex numbers, to "contain" (i.e., be mentally associated with via $\in$) infinitely many elements.

This constructive interpretation eliminates the ghost of actual, "completed" rather than "potential" infinity from the foundations of mathematics.

3.4 Reflection levels

In this section, we give a short preview of the reflection process to be described in more detail in NEUMAIER & SCHODL [57] and NEUMAIER et al. [58], since this is relevant for the understanding of the present document.

The paper [57] first defines a canonical set $\mathbb{O}_1$ of **reflected objects**; these are special objects designed to play the role of a reflected object level inside the object level, whereas the object level serves as a reflected subject level.

The reflected objects are then given the algebraic structure of a mathematical framework. The particular mathematical framework constructed in this way is called the **FMATHL mathematical framework**. It is objectively defined, exists exactly once in each subject level with a specified object level, and has in each subject level precisely the same nameable properties.

Then the notion of an expression is reflected. This enables the representation of all metaexpressions as expressions inside the object level. We obtain a set $\mathbb{E}$ of objects called **expressions**. We then reflect, in [58], a language that has all terminology needed to express the axioms for a mathematical framework and the implied definitions from the present document, so that all axioms and definitions can be represented as reflected objects that we may call **reflected axioms** and **reflected definitions**.

Therefore, we may regard $\mathbb{O}_1$ as a subject level in its own right. The original subject is still needed to perform activities in this new subject level, but its language has changed since it now uses texts formalized in $\mathbb{O}_1$ rather than in the former subject level. (See also NEUMAIER [55] for an informal exposition of such a reflection process.)

Since every subject level contains a unique FMATHL mathematical framework, $\mathbb{O}_1$ contains one, too, which we denote by $\mathbb{O}_2$. Proceeding inductively, we can construct a **hierarchy** of frameworks $\mathbb{O}_k$ ($k = 1, 2, 3, \ldots$) such that $\mathbb{O}_{k+1}$ is the unique FMATHL mathematical framework in $\mathbb{O}_k$, regarded as a subject level.

As a result, we have an infinite hierarchy of functionally equivalent object levels $\mathbb{O}_k$ nested in the helical (almost cyclical) manner characteristic for the reflection of foundations. Thus it is possible to reason objectively about anything in the subject level of relevance to the object level, by reflecting it inside the object level, and to reason objectively about this reasoning process by reflecting it, too, going as deep into the hierarchy as is needed for a particular investigation.

Helicity refers to the fact that while in each framework $\mathbb{O}_k$, regarded on its own, one has identical names and properties for every object. However, as $\mathbb{O}_{k+1}$ is contained in $\mathbb{O}_k$, the reflected objects in $\mathbb{O}_{k+1}$ also have different names and properties when considered as objects from $\mathbb{O}_k$. Of course, the differences are of exactly the same nature as those between concepts and metaconcepts.

Note that there is no level above the original subject level we started with, since this is fixed by the implementation inside the subject possessing the hierarchy of object levels. Thus, in contrast to the hierarchies constructed by TARSKI [69] or KRIPKE [43] that employ infinitely many and hence questionable language extensions, the FMATHL hierarchy is built inside a fixed subject level, hence has no associated ontological problems.

3.5 Countability

An object Z is called **countable** if there is an object f such that $Z \subseteq f[\mathbb{N}]$.

The well-known diagonal argument of CANTOR [16] shows that SUBSETS(TEXT) is not countable. Since SUBSETS(TEXT) is small by (P46), there exist uncountable sets on the object level, for example SET(SUBSETS(TEXT)).

Within our metacountable metaset of objects one can define on the object level uncountable sets such as the reals, but the concept of a countable set does not coincide with that of a metacountable set: The metaset of objects contained in the set SET(SUBSETS(TEXT)) is metacountable since it is a subset of the metacountable set $\mathbb{O}$ of all objects.

This discrepancy between the notions of countable and metacountable demonstrates an unavoidable limitation of reflection of the subject level inside the object level. It must be present in *any* formalization of mathematics in first order predicate logic, since then the Löwenheim-Skolem theorem guarantees the existence of a countable model.

The confusion that arises when one tries to equate countability and metacountability is known as **Skolem's paradox** although the discussion of these phenomena in SKOLEM [68] is devoid of any air of paradox. Indeed, Cantor's diagonal argument showing the uncountability of SUBSETS(TEXT) only amounts to showing the nonexistence of a bijection between the natural numbers and the real numbers. It says nothing about a metabijection between the corresponding metasets, which, as we just did, is easy to construct.

We conclude that there are intrinsic reasons preventing that the object level can faithfully represent the subject level; it can only reflect it in an imperfect way. As we shall see in the companion paper [57], we can achieve reflection in the sense of an isomorphic embedding of the part of the subject level needed to express the properties of the object level. But as we have seen here, we cannot achieve agreement of the concepts for arbitrary objects. This is another manifestation of the incompleteness results of Gödel and the undefinability results of Tarski.

Cardinal numbers. Note that on the basis of the above alone it is undecidable whether or not the set SUBSETS(SUBSETS(TEXT)) is small. Thus FMATHL makes no assumptions about the existence of more than two infinite cardinals, namely those of TEXT, SUBSETS(TEXT). Already the cardinal number of the set SUBSETS(SUBSETS(TEXT)), i.e., the "set" of equivalence classes of objects in bijection with SUBSETS(SUBSETS(TEXT)), cannot be formed without further assumptions.

The restricted hirarchy of sets in FMATHL resembles the "countable mathematics" of FRIEDMAN [27] and the "pocket set theory" of HOLMES [35, 36]. But FMATHL is slightly more generous and allows the continuum to be small in order that one can naturally reflect the development of context logic in NEUMAIER & MARGINEAN [56].

The motivation for this lack of commitment is that most of mathematics outside axiomatic set theory does not need more. In the few cases where it is needed, one can simply make further assumptions about sets considered to be small. We could have added an axiom that whenever Z is small then SUBSETS(Z) is small, bringing the system closer to traditional set theory, but outside of set theory itself (which, in FMATHL, has only the same status as group theory – the study of mathematical systems satisfying certain axioms) this is needed too rarely to justify assuming it at the basis of mathematics in general.

3.6 Quantifiers

In this section, we reflect quantification, to recover the usual mathematical apparatus of background notation.

We define the **quantifiers** $\forall$, $\exists$, $\exists_{\leq 1}$, and $\exists_1$ by writing, for any two existing objects Z and f,

$$\begin{array}{lll} \forall x \in Z : f(x) & \text{for} & (Z \mid f) = Z, \\ \exists x \in Z : f(x) & \text{for} & \mathrm{SET}(Z \mid f) \neq \emptyset, \\ \exists_{\leq 1} x \in Z : f(x) & \text{for} & \forall x \in Z : \forall y \in Z : f(x) = f(y) \Rightarrow x = y, \\ \exists_1 x \in Z : f(x) & \text{for} & (\exists x \in Z : f(x)) \wedge (\exists_{\leq 1} x \in Z : f(x)). \end{array}$$

Therefore, formulas with quantifiers function in FMATHL simply as mnemonic shorthand for some frequently occurring subexpressions rather than constituents from an independent background from predicate logic. The introduction of quantifiers does not change the basic propositional character of the FMATHL logic.

Their traditional meaning, apparent in the verbalization "for all $x \in Z$, we have" for $\forall x \in Z :$, "for some $x \in Z$, we have" or "there is some $x \in Z$ with" for $\exists x \in Z :$, "there is at most one $x \in Z$ with" for $\exists_{\leq 1} x \in Z :$, and "there is a unique $x \in Z$ with" for $\exists_1 x \in Z :$, can be deduced from the function axioms (P22), (P26), and (P25): If the object f is defined on the set Z, we have

$\forall x \in Z : f(x)$ iff $x \in Z$ implies $f(x) = \mathbf{1}$, (**for all reflection**)
$\exists x \in Z : f(x)$ iff $f(x) = \mathbf{1}$ for some $x \in Z$, (**exist reflection**)
$\exists_{\leq 1} x \in Z : f(x)$ iff $x, y \in Z$ and $f(x) = f(y) = \mathbf{1}$ imply $x = y$, (**at most reflection**)
$\exists_1 x \in Z : f(x)$ iff $\exists x \in Z : f(x)$ and $\exists_{\leq 1} x \in Z : f(x)$. (**unique reflection**)

Note that on the object level, we can quantify only over elements from some object. Since FMathL does *not* have a notion of unrestricted quantification ("$\forall x$" or "$\exists x$"), the FMathL logic is not quite a first order predicate logic. It could be considered as a kind of higher order logic with all objects as types (so that belonging to a type is not decidable in general), but it is most natural thought of a quantifier-free predicate logic (i.e., a propositional logic with variables for mathematical objects) of a similar nature as the ε-calculus of Hilbert & Bernays [34].

3.7 Paradoxes

In FMathL, a **paradox** is a definition from which some nominal object can be constructed. In this section, we give some formal background and then give a few typical examples.

Often, paradoxes are related to certain self-referential definitions. A **self-referential definition** is a formula

$$u := E(u), \tag{3.1}$$

where E is an expression and $E(x) = E\Big|_{z=x}()$.

Note the difference to the kinds of definitions we had so far: definition by abbreviation, which assigns a name to an object given by a formula, and definition by abstraction, which defines a function through λ-notation. In both cases, the variable whose meaning was defined did not occur on the right-hand side of the definition. In contrast, in a self-referential definition, the variable whose meaning is defined appears both on the left-hand side and on the right-hand side. As a result, a self-referential definition entails the *assumption* that an object with the property $u = E(u)$ exists. Whether or not this assumption is justified must be investigated by arguments, leading to so-called fixed-point theorems.

An object u satisfying $u = E(u)$ is called a **fixed point** of the expression E with respect to the substituted variable z. In general, there may be no, just one, or several fixed points. The expression E is called **fixed-point free** with respect to the substituted variable z if no such fixed-point exists.

Clearly, the self-referential definition $u := E(u)$ makes sense only when E has a fixed point. But it is easy to find expressions that have no fixed points; thus corresponding self-referential definitions may define at best a nominal object. An example is the expression $z = \mathbf{0}$, where the corresponding self-referential definition leads to the **liar paradox** $L := (L = \mathbf{0})$ discussed in Section 2.5.

Under certain assumptions, the existence of a fixed point can be established, however, constructively; this is possible when the self-referential definition has a **recursive** interpretation in terms of a well-founded relation, in which case an inductive existence proof can be given. As an example we prove here the following recursion theorem.

3.7.1 Theorem. *For existing objects Z and e, let E be an expression such that $E(x,n) \in Z$ for all $x \in Z$ and $n \in \mathbb{N}$. Then there is a unique function $f : \mathbb{N} \to Z$ such that*

$$f(1) = e,$$

$$f(n+1) = E(f(n), n) \quad \textit{for } n \in \mathbb{N}.$$

Proof. The object

$$X := \{Y \subseteq \mathbb{N} \times Z \mid (1,e) \in Y, \forall t \in Y : (t_1 + 1, E(t_2, t_1) \in Y\}$$

contains the object $\mathbb{N} \times Z$. Hence $\textsc{Intersect}(X)$ is nonempty, and

$$f(x) := \textsc{Choice}(\{t_2 \in Z \mid t \in \textsc{Intersect}(X), t_1 = x\}$$

defines a function with the required property. Uniqueness follows easily by induction. □

One generally refers to the function f constructed in the proof as the function f defined by

$$f(1) := e,$$

$$f(n+1) := E(f(n), n) \quad \text{for } n \in \mathbb{N}.$$

On the other hand, if the self-referential definition is intrinsically **circular**, i.e., cannot be interpreted as a recursion, the status of the defined variable is intrinsically unclear. An example is the trivial self-referential definition $L := L$, which says nothing at all about L. But in some circular cases, namely for paradoxes such as the liar paradox, one can definitely prove that a corresponding object, if it exists, must be nominal.

Yablo's paradox. As an example of a nontrivial self-referential definition we consider the paradox of Yablo [73] (1993). It can be formalized as the definition

$$S := \lambda k.(\forall j \in \mathbb{N} : j > k \Rightarrow \neg S(j)). \tag{3.2}$$

This formalization shows that the paradox involves self-reference on the level of the sequence S, contrary to the title of Yablo's paper. (His informal version superficially looks free of self-reference since the sequence is mentioned only in passing.)

Assuming that S exists, one derives a contradiction as follows: $S(k)$ holds iff $S(j)$ is false for all natural numbers $j > k$. If $S(k)$ holds for some natural number k then $\neg S(j)$ for all natural numbers $j > k$, but, upon ignoring the case $j = k+1$, this implies that $S(k+1)$ holds, contradiction. Thus $S(k)$ is false for all natural numbers. But this implies that $S(1)$ holds, contradiction. Therefore S must be nominal.

Hilbert's paradox. We now consider a paradox by HILBERT [33] (1905) (cf. KAHLE [37]). By our previous theory, the object $\lambda x.(x\,@\,x = \mathbf{0})$, i.e., the function H defined by

$$H(x) := (x\,@\,x = \mathbf{0}),$$

is well-defined in FMATHL. If we evaluate H at the argument $x = H$, we find that $H(H) = (H(H) = \mathbf{0})$, a formula that is a variant of the liar paradox (2.9). The argument given there implies that $L := H(H)$ is nominal, and hence $H(H)$ is nominal since by definition, H is a function.

In the unrestricted λ-calculus with identity (which Hilbert implicitly employed although the concept was formally defined only much later by CHURCH [18]), we would have obtained a contradiction.

Hilbert's paradox is a functional variant of the well-known set-theoretic paradox by Russell. Indeed, if we identify in naive set theory a set R with its characteristic function r, satisfying $r(x) = \mathbf{1}$ if $x \in R$ and $r(x) = \mathbf{0}$ otherwise, the Russell set $R := \{x \mid x \notin x\}$ (a formula we cannot form in FMATHL) becomes $R = \{x \mid x(x) = \mathbf{0}\}$, and its characteristic function is H. In this sense, H is the characteristic function of the Russell set.

Hilbert's paradox shows that not all paradoxes have a self-referential nature; self-application may have a similar effect. Indeed, as any fixed-point free expression E leads to a paradox with the self-referential definition $u := E(u)$, so it leads to a paradox with the following self-applicative but not self-referential construction.

3.7.2 Theorem. *Suppose that E is fixed-point free, and let f be the object defined by*

$$f(x) := E(x(x)).$$

Then $f(f)$ is nominal.

Proof. If $x := f(f)$ exists then f exists, and $x = f(f) = E(f(f)) = E(x)$, contradicting the assumption. Hence $f(f)$ does not exist and is nominal. □

Bibliography

[1] American Mathematical Society, A Special Issue on Formal Proof, Notices Amer. Math. Soc. 55 (2008).
`http://www.ams.org/notices/200811/`

[2] Aristoteles, Ta meta ta physika VIII, 1042a21, Athens, ca. 350 B.C.
`http://remacle.org/bloodwolf/philosophes/Aristote/tablemetaphysique.htm`
English translation: Aristotle, Metaphysics (transl. by W.D. Ross).
`http://classics.mit.edu/Aristotle/metaphysics.html`
`http://www.aristoteles-heute.de/sein_a/sein_A/aristoteles/metaphysics/ftp/Metaphysics.htm`

[3] Aristoteles, Categoriae 2a11-19, Athens, ca. 350 B.C.
`http://remacle.org/bloodwolf/philosophes/Aristote/tablemetaphysique.htm`
English translation: Aristotle, Categories (transl. by E.M. Edghill).
`http://www.aristoteles-heute.de/sein_a/sein_A/aristoteles/categories/ftp/Categories.htm`

[4] S. Awodey, A brief introduction to algebraic set theory, Preprint (2008).
`http://www.andrew.cmu.edu/user/awodey/preprints/astIntroFinal.pdf`

[5] S. Awodey, From Sets to Types to Categories to Sets, Preprint (2009).
`http://www.andrew.cmu.edu/user/awodey/preprints/stcsFinal.pdf`

[6] H. Barendregt, Foundations of Mathematics from the Perspective of Computer Mathematics, Festschrift for Bruno Buchberger (P. Paule, ed.), to appear (2009).
`ftp://ftp.cs.kun.nl/pub/CompMath.Found/buchberger.pdf`

[7] H. Barendregt, M. Bunder and W. Dekkers, Systems of illative combinatory logic complete for first-order propositional and predicate calculus, J. Symbolic Logic 58 (1993), 769–788.

[8] H. Barendregt and F. Wiedijk, The Challenge of Computer Mathematics, Phil. Trans. Roy. Soc. A 363 (2005), 2351–2375.
`http://www.cs.ru.nl/F.Wiedijk/pubs/rspaper.pdf`

[9] M.J. Beeson, Towards a computation system based on set theory, Theor. Computer Sci. 60 (1988), 297–340.

[10] M.J. Beeson, The mechanization of mathematics, pp. 77–134 in: Alan Turing: Life and Legacy of a Great Thinker (C. Teuscher, ed.), Springer, Berlin 2003.

[11] P. Benacerraf, What Numbers Could Not Be, The Philosophical Review 74 (1965), 47–73.

[12] E. Bishop, Foundations of constructive analysis, McGraw-Hill, London 1967.

[13] N. Bourbaki, Théorie des Ensembles, Hermann, Paris 1970. English translation: Theory of sets, Hermann, Paris 1968.

[14] E.D. Buckner, The Myth of Ockham's razor, in: The logic museum, Web document (2006).
`http://uk.geocities.com/frege@btinternet.com/latin/mythofockham.htm`

[15] M.W. Bunder, A one axiom set theory based on higher order predicative calculus, Arch. Math. Logik Grundl. 23 (1983), 99–107.

[16] G. Cantor, Über eine elementare Frage der Mannigfaltigkeitslehre, Jahresber. Deutsch. Math.-Verein. 1 (1890-1), 75–78.

[17] D. Carlisle, P. Ion and R. Miner, Mathematical Markup Language (MathML) Version 3.0, Web document.
`http://www.w3.org/TR/2009/WD-MathML3-20090604/`

[18] A. Church, A set of postulates for the foundations of logic, Ann. Math. 33 (1932), 346–366; Ann. Math. 34 (1933), 839–864

[19] H.B. Curry, The combinatory foundations of mathematics, J. Symbolic Logic 7 (1942), 49–64.

[20] H.B. Curry and R. Feys, Combinatory logic, Vol. 1, North-Holland, Amsterdam 1958.

[21] Epimenides, ca. 600 B.C.
Diels-Kranz: Die Fragmente der Vorsokratiker, 6. Auflage 1952, I 3B1, Epimenides-Zitat aus Tit. 1,12.

[22] S. Feferman, Toward useful type-free theories. I, J. Symbolic Logic 49 (1984), 75–111.

[23] S. Feferman, Reflecting on incompleteness, J. Symbolic Logic 56 (1991), 1–49.

[24] S. Feferman, Axioms for determinateness and truth, The Review of Symbolic Logic 1 (2008), 204–217.

[25] W. Feit and J.G.Thompson, Solvability of groups of odd order, Pacific J. Math. 13 (1963), 775–1029.

[26] A. Fraenkel, Über den Begriff "definit" und die Unabhängigkeit des Auswahlaxioms, Sitz. Ber. Preuss. Akad. Wiss. (Math. Klasse) (1922), 253–257. English translation: The notion of 'definite' and the independence of the axiom of choice, pp. 284–289 in: From Frege to Gödel: A Source Book in Mathematical Logic 1879–1931 (J. van Heijenoort, ed.), Harvard Univ. Press. 1967.

[27] H.M. Friedman, Adventures in the foundations of mathematics, Lecture 1: The structure of proofs, VIGRE Lectures, Ohio State University 2003.
`http://www.math.ohio-state.edu/~friedman/pdf/VIGRE050903.pdf`

[28] R.L. Griess, Jr, The Friendly Giant, Inv. Math. 69 (1982), 1–102.

[29] T. Forster, Implementing mathematical objects in set theory, Logique et Analyse, to appear (2009).

[30] K. Gödel, Über die Vollständigkeit des Logikkalküls, Dissertation, Universität Wien 1929.

[31] K. Gödel, Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I, Monatshefte für Mathematik und Physik 38 (1931), 173–198. Engl. translation (by M. Hirzel): On Formally Undecidable Propositions Of Principia Mathematica And Related Systems, Boulder 2000.
`http://www.research.ibm.com/people/h/hirzel/papers/canon00-goedel.pdf`

[32] N. Goodman and J. Myhill, Choice implies excluded middle, Zeitschr. f. Math. Logik und Grundlagen d. Math. 24 (1978) 461.

[33] D. Hilbert, Logische Principien des mathematischen Denkens, Vorlesung Sommersemester 1905, Ausarbeitung von Ernst Hellinger. Bibliothek des mathematischen Instituts der Universität Göttingen, 1905.

[34] D. Hilbert and P. Bernays, Grundlagen der Mathematik, 2 Vols., Springer, Berlin 1934, 1939.

[35] M.R. Holmes, Pocket set theory, Manuscript (2007).
`http://math.boisestate.edu/~holmes/holmes/pocket.dvi`

[36] M.R. Holmes, Alternative set theories, Manuscript (2006).
`http://math.boisestate.edu/~holmes/holmes/separticle.pdf`

[37] R. Kahle, David Hilbert über Paradoxien, Pré-Publicações do Departamento de Matemática, Universidade de Coimbra, Preprint Number 06-17 (2006)
`http://www.mat.uc.pt/preprints/ps/p0617.pdf`

[38] The Bible, King James Version, 1611.
http://www.ccel.org/ccel/bible/kjv.i.html

[39] K. Kofler, P. Schodl, and A, Neumaier, Limitations in Content MathML, Manuscript (2009).
http://www.mat.univie.ac.at/~neum/FMathL/mathml0.pdf

[40] N. Kolev, Generating Proof Representation Structures in the Project NAPROCHE, Master thesis, University of Bonn, 2008.
http://www.math.uni-bonn.de/people/naproche/

[41] S. Kleene, Introduction to metamathematics, Van Nostrand, Princeton, N.J., 1952.

[42] S. Kleene and J.B. Rosser, The inconsistency of certain formal logics, Annals of Math. 36 (2) (1935), 630–636.

[43] S. Kripke, Outline of a Theory of Truth, J. Philosophy 72 (1975), 690–716.

[44] L. Kronecker, Berlin meeting of the Vereinigung deutscher Naturforscher und Ärzte, 1886. Quoted after H. Weber, Leopold Kronecker, Jahresber. DMV 2 (1892), 25–31.

[45] D. Kuehlwein, A Short Introduction To Naproche v0.1, Manuscript (2008).
http://www.math.uni-bonn.de/people/naproche/

[46] F.W. Lawvere, An elementary theory of the category of sets, Proc. Nat. Acad. Sci. USA 52 (1964), 1506–1511.
An elementary theory of the category of sets (long version) with commentary, Repr. Theory Appl. Categ. 12 (2005), 1–35.

[47] F.W. Lawvere, The category of categories as a foundation for mathematics pp. 1–21 in: Proc. Conf. on Categorical Algebra, La Jolla, 1965, (S. Eilenberg et al., eds.), Springer, Berlin 1966.

[48] H. Leitgeb, What theories of truth should be like (but cannot be), Philosophy Compass 2 92007), 276–290.

[49] L. Löwenheim, Über Möglichkeiten im Relativkalkül, Mathematische Annalen 76: 447–470.

[50] M. Makkai, Towards a categorical foundation of mathematics, pp. 153–190 in: Logic Colloquium '95 (J.A. Makowsky and E.V. Ravve, eds.), Springer, Berlin 1998.

[51] C. McLarty, Learning from questions on categorical foundations, Philosophia Mathematica (III) 13 (2005), 44–60.

[52] C. McLarty, Introduction to Categorical Foundations for Mathematics, version of August 14, 2008.
http://www.cwru.edu/artsci/phil/Roskilde*school-814.pdf (with * replaced by a blank %20)

[53] A. Neumaier, Analysis und lineare Algebra, Lecture Notes in German (2008). `http://www.mat.univie.ac.at/~neum/FMathL/ALA.pdf`

[54] A. Neumaier, FMathL – Formal Mathematical Language, Web site (2009). `http://www.mat.univie.ac.at/~neum/FMathL.html`

[55] A. Neumaier, Mathematical reasoning - a bottom-up approach, Web document (2009). `http://www.mat.univie.ac.at/~neum/FMathL/reasoning.html`

[56] A. Neumaier and F.A. Marginean, Logic in context, Manuscript (2009). `http://www.mat.univie.ac.at/~neum//FMathL/context.pdf`

[57] A. Neumaier and P. Schodl, Constructing the FMathL mathematical framework, In preparation (2009).

[58] A. Neumaier, P. Schodl, and K. Kofler, Specifications in the FMathL mathematical framework, In preparation (2009).

[59] J. von Neumann, Eine Axiomatisierung der Mengenlehre, J. reine angew. Math. 154 (1925), 219–240. English translation: An Axiomatization of Set Theory pp. 394–413 in: From Frege to Gödel (J. van Heijenoort, ed.), Harvard University Press, Cambridge, MA 1967.

[60] William of Ockham, Summa Totius Logicae I, ca. 1323. English translation: M.J. Loux, Ockham's Theory of Terms: Part I of the Summa Logicae, University of Notre Dame Press, Notre Dame, IN, 1974.
An online translation of Cap. 10 is at `http://uk.geocities.com/frege@btinternet.com/connotation/ockhamconnote.htm`

[61] R. Penrose, Shadows Of The Mind, Oxford Univ. Press 1994.

[62] The QED manifesto, pp. 238–251 in: Automated Deduction - CADE 12, Lecture Notes in Artificial Intelligence, Vol. 814, Springer, Berlin 1994. `http://www.cs.ru.nl/~freek/qed/qed.html`

[63] Plato, Hippias meiẑon 304a, ca. 390 B.C.. `http://www.perseus.tufts.edu/cgi-bin/ptext?lookup=Plat.+Hipp.+Maj.+281a` (English translation, with link to Greek original)

[64] Plato, Hippias Elattôn 376b, ca. 390 B.C.. `http://www.perseus.tufts.edu/cgi-bin/ptext?lookup=Plat.+Hipp.+Min.+376b` (English translation, with link to Greek original)

[65] B. Russell, Recent work in the philosophy of mathematics, The International Monthly, 1901. Reprinted as "Mathematics and the Metaphysicians", pp. 75–95 in: Mysticism and Logic, Allen and Unwin, London 1917.

[66] M. Schönfinkel, Über die Bausteine der mathematischen Logik, Math. Annalen, 92 (1924), 305–316. English translation: On the building blocks of mathematical logic, pp. 355–366 in: From Frege to Gödel: A Source Book

in Mathematical Logic (J. van Heijenoort, ed.), Harvard University Press, Cambridge, MA, 1967.

[67] J.P. Seldin, The logic of Curry and Church, in: Logic from Russell to Church, Handbook of the History of Logic, Vol. 5 (D.M. Gabbay and J. Woods, eds.), Elsevier, 2009
http://people.uleth.ca/~jonathan.seldin/CCL.pdf

[68] T. Skolem, Einige Bemerkungen zu axiomatischen Begründung der Mengenlehre, Mathematikerkongressen i Helsingfors 4.-7. Juli 1922, Den femte skandinaviska matematikerkongressen, Redogoerelse, 217–232. English translation: Some remarks on axiomatized set theory, pp. 290–301 in: From Frege to Gödel: A Source Book in Mathematical Logic 1879–1931 (J. van Heijenoort, ed.), Harvard Univ. Press. 1967.

[69] A.Tarski, Pojecie prawdy w jezykach nauk dedukcyjnych, Towarzystwo Naukowe Warszawskie, Warszawa 1933. German translation: Der Wahrheitsbegriff in den formalisierten Sprachen, Studia Philosophica 1 (1935), 261–405. English translation: The Concept of Truth in Formalized Languages, pp. 152-277 in: A. Tarski, Logic, Semantics and Metamathematics, Oxford Univ. Press 1956.
A discussion of Tarski's undefinability theorem is in http://en.wikipedia.org/wiki/Tarski's_undefinability_theorem

[70] W.P. Thurston, On proof and progress in mathematics, Bull. Amer. Math. Soc. (N.S.) 30 (1994) 161–177.
http://arxiv.org/abs/math.HO/9404236

[71] F. Wiedijk, The QED Manifesto Revisited, pp. 121–133 in: From Insight to Proof, Festschrift in Honour of Andrzej Trybulec (R. Matuszwski and A. Zalewska, eds.), Studies in Logic, Grammar and Rhetoric 10, University of Bialystok, 2007.
http://www.cs.ru.nl/~freek/pubs/qed2.pdf

[72] L. Wittgenstein. Tractatus logico-philosophicus, Annalen der Naturphilosophie 14 (1921). English translation by D. Pears and B. McGuinness, Routledge 1961.
http://www.gutenberg.org/etext/5740

[73] S. Yablo, Paradox without self-reference, Analysis 53 (1993), 251–252.

[74] E. Zermelo, Untersuchungen über die Grundlagen der Mengenlehre I, Math. Annalen 65 (1908), 261–281. English translation: Investigations in the foundations of set theory I, pp. 199–215. in: From Frege to Gödel: A Source Book in Mathematical Logic 1879–1931 (J. van Heijenoort, ed.), Harvard Univ. Press. 1967.

Index