

# 0-based Logic

## Introduction

The pursuit of knowledge has always involved re-examining our fundamental beliefs. From the realisation that the Earth is not flat to the acceptance of a relativistic universe, history shows that progress results from challenging assumptions.

This paper presents the result of five years of meticulous thought on the relationship between logic, infinity, and reality. It identifies a critical logic error in mathematics and physics and proposes a solution to unify science, mathematics, and computing under a clarified definition of logic. This correction also enables the development of 3R, which provides clearer and more consistent logical modelling.

## Re-factoring

In programming, a minor oversight can lead to a critical bug as software complexity increases. Similarly, our established mathematical and physical models rest upon a fundamental misunderstanding. We have confused physics with mathematics. We do not need to dismantle and rebuild everything; we simply need to restore foundational integrity by clearly defining logic.

Like using version control in software: instead of discarding the entire codebase, we revert to a build better aligned with unambiguous logic (as defined by computing). For engineers and programmers, this is akin to updating a critical utility library your applications depend on. This update enhances the precision and efficiency of the underlying algorithms without overtly changing everyday calculations or analyses.

This correction clarifies the foundational logic behind our models, ensuring consistency across science, mathematics, and computing. It sets the stage for a unified approach to logical reasoning.

This work stems from a personal journey shaped by a childhood addiction to computer programming and mathematics—a condition that has remained untreated for over 40 years. It prepares the ground for the next generation of AI—sophisticated, beneficial, and safe. Logically consistent AI is not a threat to humanity; rather, it will help us understand and curb our evolved, self-interested behaviours, which are the real threat. The human brain, a product of natural selection, prioritises emotional desires over logical thought patterns, unlike our silicon helpers, designed to work tirelessly and selflessly on our behalf.

## Axioms

The early 20th century marked significant scientific progress but also a regressive step in mathematics, sowing seeds of confusion within the scientific community. Fortunately, the advent of computing in the 1950s provided an opportunity to reverse this step and clarify the confusion.

Understanding why this error occurred is crucial to preventing its recurrence. Our brains, products of evolution, are honed for survival, not for logic and mathematics. Emotion-driven and self-interest focused, our thought processes evolved to prioritise survival over abstract reasoning.

Historically, beliefs have framed our understanding of the world, evolving from religious convictions to axioms, considered self-evident truths before the 20th century. Words like **true** and **false** are abstract language and thus inherently political, rooted in axioms or beliefs. A computer has no understanding of **true** until we define a boolean class and specify the meaning of the word.

A significant shift occurred with the mathematician Cantor in the early 20th century. He redefined mathematics in a way that is inconsistent with what we subsequently defined as computational logic. This pivotal error prompted my five-year investigation into Cantor's work. For those outside mathematics, the intricacies of infinite set theory may seem obscure. It is often assumed that mathematics and computing are seamlessly aligned within the same logical framework; however, this is not the case.

In the 21st century, we must move beyond axioms. Misinterpreting the abstract definitions of logic or natural language as reality can lead to profound confusion. Beliefs and assumptions have no place in science. We are now positioned to define logic and mathematics with precision, free from axioms, enabling us to logically model reality and empirically test our models.

## 0-based Logic

Non-axiomatic logic, starts with a single base symbol to ensure non-ambiguity and consistency. The initial symbol is 0, chosen for its universal recognition. Within logic, 0 initially has no intrinsic meaning but serves as the foundation for all definitions.

1 is then defined as  $\neg 0$ . This step defines both 1 and the  $\neg$  operator. While the natural language word **not** has a similar meaning to  $\neg$ , **not** is an axiomatic word. We are defining a non-axiomatic language, which has no natural language assumptions. We are however free to subsequently define the word **not** within a programming language.

Defining symbols is the essence of creating logic. It does not assume but establishes meaning. 0 was initially meaningless. After defining 1 we have three symbols with a clearly defined relationship. Each new definition adheres to one strict rule: any new definition may neither directly nor indirectly introduce any contradiction into the language. With the base definition as  $1 = \neg 0$ , the primary rule of logic is to disallow any language construct which permits  $1 = 0$ .

0-based (non-axiomatic) logic does not ignore existing mathematical knowledge but validates it. Mathematics is strengthened with explicit and precise logic. This respects historical discoveries and theorems while placing them within a language free from assumptions. Mathematical progress is honoured, whilst ensuring logical constructs are consistent and non-ambiguous.

Logic is clearly embodied in computing systems, where precise and rigorous logic is crucial. Each command in a computer's architecture is meticulously accurate, eliminating ambiguity. 0-based logic is not just theoretical but a practical reality in computer programming. Logic gates, particularly NAND gates, demonstrate that 0-based logic is not only feasible but already in use. While hardware and software design sometimes relaxes strict consistency rules, these deviations are intentional. Less stringent elements are sometimes introduced for practical reasons, knowing that the system's foundation is precisely defined. In logic, ambiguity has no place; each gate, circuit, and code line can trace its origin to the base symbol 0.

## Definition

Consider the binary sequence: 0, 1, 10, 11, 100, 101... At first glance, axiomatic intuition sees a familiar counting pattern, taught from early childhood. However, to truly understand the essence of logic, we must reverse our thinking. It's not that this sequence represents counting; rather, this sequence **defines** counting within the language of logic.

In 0-based logic, counting does not derive its meaning from an external belief. Instead, counting is defined by this binary sequence. It's a foundational building block, not a derivative concept. Thus, the binary sequence that we instinctively recognise as counting is actually a prime example of logic at work. Through this sequence, we understand what it means to count. Counting within logic isn't axiomatic—it is **defined**.

Highlighting this distinction is critical to understanding the departure from traditional axiomatic systems. Where axioms are **assumed**, here every concept is deliberately and rigorously **defined**. This paradigm shift is central to constructing logic that is consistent and unambiguous, mirroring the stringent requirements of computing, our most powerful logical tool.

It is worth noting that when we articulate the binary sequence 0, 1, 10, 11, 100, 101... in conversation or traditional mathematical notation, we are not adhering to the stringent standards of rigour that a computer requires. Rigour in computing means constructing definitions so precisely that there can be no misinterpretation or ambiguity.

Turning such a sequence into a rigorously defined construct is indispensable for building computers and

computational models. Each step must be articulated in terms of logical operations, generally through code but ultimately through the use of foundational operators such as NAND.

For human communication, imposing such rigorous definitions in every explanation can become burdensome and impede understanding. It's not always necessary to descend to the level of machine code to validate our explanations or understandings. Human to human explanations generally require far less rigour. In human terms, explanations must balance clarity with precision; in computing, it is the unyielding consistency which creates clarity.

## $\infty$ and $\langle 1|0 \rangle$

In an axiomatic system, logical incompleteness is often interpreted through the lens of Gödel's incompleteness theorems, which state that within any sufficiently complex set of axioms, there are propositions that cannot be proven. This interpretation is contingent upon a series of assumed axioms.

In contrast, 0-based logic provides a definitive approach. It defines incompleteness not as a shortcoming in proving certain axiomatic truths but as an inherent property of logic. Consider the sequence that defines counting within 0-based logic: 0, 1, 10, 11, 100, 101, ... This sequence is inherently designed to extend indefinitely, not due to an externally imposed rule but because its logical definition dictates a never-ending process. The essential aspect here is that our sequence does not end because it is constructed from a consistently applied logic—it is always possible to derive the next element. This unending extension is a consequence of a system designed to be inherently expansive. Thus, within 0-based logic, the lack of an end to counting is not a flaw but a testament to a system that is fundamentally consistent. Axiomatic words like **unbounded** or **infinite** emerge naturally to describe the observable characteristics of the system at work. Logical incompleteness in this context is not a deficiency but an intrinsic property of a sequence defined to progress indefinitely.

$\infty$  in 0-based logic is defined as the theoretical end-point of the logical sequence: 0, 1, 10, 11, 100, 101, ... However, this sequence is logically inconsistent when extended to  $\infty$  because it fails to account for the inherent incompleteness. To achieve logical consistency, logic must define uncertainty in the form of  $\langle 1|0 \rangle$ . This notation defines 1,0 super-position, representing the inherent uncertainty within a logical system. The extended sequence can thus be represented as 0, 1, 10, 11, 100, 101, ...,  $\langle 1|0 \rangle$ ,  $\infty$ . This sequence is logically consistent because it integrates the concept of uncertainty before reaching  $\infty$ .

To incorporate  $\infty$  without undermining consistency, logic defines a rigorous set of rules traditionally known as **analytic reasoning**. These rules guide  $\infty$ 's integration into logical operations to prevent contradictions. The analytic rules of calculus, which oversee  $\infty$ , are logically validated to demonstrate how they maintain consistency. As a result,  $\infty$  acts as a conduit between the finite and the unbounded, providing a logical way to express and contemplate the infinite. It recognises that our logical language is broader and more intricate than purely finite, inviting us to explore this expanded logical space with thoughtful precision.

## Fixing Axiomatic Errors

The concept of **edge cases** derived from computing is crucial for grasping the essence of logic. Consider the closed interval  $[0,1]$ ; its boundaries are quite clear. In contrast to axiomatic logic, 0-based logic interprets this interval definitively. Declaring  $[0,1]$  immediately points to the values 0 and 1, but what lies between these values is not clearly defined. Introducing algebra into our dialogue of logic mirrors the introduction of variables in computing. Here, variables are the means to define  $\langle 1|0 \rangle$  uncertainty while upholding logical consistency and precision.

Within logic, the definition  $0 \leq x \leq 1$  may be employed to specify a state of logical superposition that straddles 0 and 1 without pinpointing a precise value. Acknowledging  $\langle 1|0 \rangle$  aids in demystifying concepts that may seem perplexing under axiomatic logic. Such was the confusion surrounding analytic reasoning, irrational numbers, and the continuum that led to Cantor's errors.

Re-examining the diagonal argument through the lens of 0-based precision reveals its flaws. Applying the argument to the sequence 0.1, 0.01, 0.11, 0.001, ... indicates the omission of the value 0.00111... However, a thorough understanding unveils that this value, analytically addressed, equates to 0.01—a number certainly not

absent from the sequence. Here lies **the countable infinity contradiction** where an attempt to enumerate  $\infty$  contravenes the incompleteness requirement of logic. The convergence towards 0.01 is not disputed; rather, it's the interpretation that shifts, reshaping perception of irrational numbers. A closer inspection of bicimal patterns helps in understanding more clearly.

Within logic, any well-defined value within the range  $[0,1]$  has a well-defined bicimal expansion. When confronted with potential ambiguity, we use the non-analytic value over its analytic convergence. So, for example, we use 0.01 rather than 0.00111... despite the convergence of the latter to the former. We may pad any value with as many trailing 0s as we wish and indeed are encouraged to do so where we wish to expose a specific level of precision in calculations.

Rational values in logic are simply those values delivered by the consistently defined division operator. As with traditional approaches, 0-based logic confirms such a constraint will cause the values to either terminate or repeat in a simple way. The value  $1/3$  in bicimal is, for example, 0.010101... equivalent to 0.333... in decimal. Clearly, we may form a multitude of other more complex never-ending patterns. The bicimal value 0.1010010001... for example, is guaranteed to deliver a value which the division operator alone cannot quite replicate. We may deliver a never-ending variety of never-ending bicimal expansions with ever more complex patterns, but to claim that this is a bigger version of  $\langle 1|0 \rangle$  is clearly misleading. Indeed, the simple bicimal sequence 0.1, 0.01, 0.11, 0.001, ... is sufficient to deliver  $[0,1]$  could we complete the sequence since it is systematically itemising all of the options. This sequence delivers what the slightly confused axiomatic language of set theory describes as a dense set.

Care should be taken if adopting set theory terminology since it fails to honour the  $\langle 1|0 \rangle$  requirement of logically defined  $\infty$  and thus introduces highly irregular naming. Countable  $\infty$  is by definition a contradiction within 0-based logic. Particular care should be taken with infinite mappings since what set theory claims to be a **bijection**, 0-based logic recognises as simply perpetually **incomplete**. Particular care should be taken when considering **real numbers** since number is abstract not real so the name is clearly misleading. The naming actually hides a mistaken belief which lies at the heart of axiomatic mathematics, namely the desire to construct logic out of reality and reality out of logic. We can be confident this is not possible. This recognition does not, however, remove the power of logic. 0-based logic understands abstract incompleteness and has crafted tools and the special symbols  $\infty$  and  $\langle 1|0 \rangle$  to deal with it.

To understand the subtle difference between axiomatic and 0-based logic, it helps to consider the interval  $[0,1]$ . We may define a variable  $x$  such that  $0 \leq x \leq 1$ . If we add the constraint that the value must be an integer, then our  $x$  may only take one of two values: 0 or 1. These various definitions are what computing describes as types. By specifying the type of our variable, we add constraints with respect to what is possible.

If we now consider the open interval  $(0,1)$ , the picture is a little more interesting. Defining  $0 < x < 1$  makes very little difference if  $x$  is left unconstrained. Indeed, we can use analytic  $\infty$  to describe this difference. Left unconstrained the interval has infinitely many distinct values since the sequence 0, 0.1, 0.01, 0.11, 0.001, ... continues indefinitely. The open interval is just 2 values smaller than the closed interval, so differs by only  $2/\infty$  which analytically is so close to 0 that we may treat it as 0. It is analytically consistent to treat both of these intervals as of length 1 provided we place no additional constraints on  $x$ . Place an integer constraint and the picture changes dramatically. Our  $x$  can now take no values at all. Both axiomatic and 0-based logic agree that  $0 < x < 1$  has no clearly defined smallest or largest value. The difference is that 0-based logic is able to explain this analytically and consistently in terms of  $\langle 1|0 \rangle$ .

### 3R

Logical edges are an unavoidable consequence of consistent, non-ambiguous logical language, where 0 is defined as  $-1$ . This inherent characteristic means that any two numbers have gaps between them since the bicimal expansion of any two numbers is always distinct; hence, any two numbers must always differ at some bicimal level. Science, however, can never measure to these levels. We must always introduce  $\langle 1|0 \rangle$  uncertainty well before the end of the bicimal expansion, typically achieving no better than nine decimal significant figures.

This limitation can be addressed by modelling these uncertainties in terms of relativistic spheres, where the spheres are designed to have no logical edges. For instance, a time-distance-force 3R sphere may be used to model atoms or the universe. This approach ensures that even where uncertainty is inherent, the model remains logically consistent and unambiguous. We have role confusion between physics and logic: three rela-

tivistic dimensions and quantum uncertainty (as modeled in Quantum Mechanics with complex numbers) are consequences of logical modelling, not strictly properties of reality. We only ever understand reality through a model however, and a 3R model is the best possible logical model. This is why the brain evolved to model in three dimensions. For more detailed information on consistent modelling of reality see 3R Scientific Modelling.

By embracing the concept of 3R relativistic spheres, we can achieve a more accurate representation of reality that aligns with our logical foundations. This method bridges the gap between the theoretical precision of logical language and the practical limitations of scientific measurement, providing a coherent framework for understanding and modelling the complexities of the universe.