Anticipation as Prediction in the Predication of Data Types

M. A. Heather*, B. N. Rossiter**
* Ambrose Solicitors, St Bede's Chambers, Jarrow NE32 5JB
michael.heather@cantab.net
**School of Computing, Engineering and Information Sciences,
Northumbria University, NE1 8ST, UK
nick.rossiter@unn.ac.uk

Abstract Every object in existence has its type. Every subject in language has its predicate. Every intension in logic has its extension. Each therefore has two levels but with the fundamental problem of the relationship between the two. The formalism of set theory cannot guarantee the two are co-extensive. That has to be imposed by the axiom of extensibility, which is inadequate for types as shown by Bertrand Russell's ramified type theory, for language as by Henri Poincaré's impredication and for intension unless satisfying Port Royal's definitive concept.

An anticipatory system is usually defined to contain its own future state. What is its type? What is its predicate? What is its extension? Set theory can well represent formally the weak anticipatory system, that is in a model of itself. However we have previously shown that the metaphysics of process category theory is needed to represent strong anticipation.

Time belongs to extension not intension. The apparent prediction of strong anticipation is really in the structure of its predication. The typing of anticipation arises from a combination of δ and μ — respectively (co) multiplication of the (co)monad induced by adjointness of the system's own process. As a property of cartesian closed categories this predication has significance for all typing in general systems theory including even in the definition of time itself.

Keywords: typing, predication, adjointness, monad, time.

1 Typing

Typing is an essential feature of modern systems theory. It is even at the essence of anticipation in anticipatory systems. Yet typing appears in many different guises. From the quest of ancient philosophers seeking the basic type of matter that makes up the world to the current theory of the Standard Model of the fundamental particles of this universe, and experiments to find them with CERN's Hadron collider – typing still remains the burning question. Modern interest in information systems, where typing has proved to be important, has confirmed recognition that physical systems and information systems are not independent of one another. However, there is a vast unknown chasm of understanding between fundamental particle types and the operational information of genome types that are needed in current biology and medicine.

1.1 Prediction and Predication Compared

An anticipatory system is one that is able to predict its own future state either from a model of itself that it has within (usually termed weak anticipation [8, 9, 12]) or from its own operation (known as strong anticipation). This types the anticipatory system. The classical Greek for type is κατηγορία (categoria), that is *category* coined by Aristotle from the legal term to indict or assert. In Latin the Romans used the word *predicamentum* from the same root as *prædĭcare* to assert from which we derive the word predicate: that which is asserted. Thus a predicate provides an attribute for a subject. However *prædĭcare* to assert is not to be confused with *predīcere* meaning to predict or foretell [1]. Thus the words *categories* and *predicates* have continued to the present day in this sense in English. The predicate is usually language-oriented and the category logic-related. Both terms are equivalent to ‘type’ although predicate is normally used exclusively for the extension of a type and a category usually for the intension. These two concepts come together in the anticipatory system whose own predicate is to predict itself.

2 Historical development of Typing

Typing can be traced back to Sanskrit literature and did not begin with Aristotle but the only pre-socratic attempts at classification relevant to anticipatory systems are perhaps the well known fundamental types of Parmenides (“everything stays the same”) and Heraclitus (“all is flux”) which still continue today to receive considerable attention in philosophy. At first sight the constancy of Parmenides may seem to have little relevance to anticipatory systems if their prime significance is to predict their own future states. However this is to presuppose a simple linear time scale as an independent variable where an anticipatory system embodies a model of itself, which is but weak anticipation. However the Parmedian aspect of anticipatory systems is their intensional form where the prediction can be ascertained from the typing of the system independently of time. The extensional form of the anticipatory system is that of Heraclitus where the anticipation lies in the semantics. The extension may well include time but again if time is an independent variable it is still only weak anticipation. In strong anticipation time is part of the data [13] and may therefore consist of a variety of forms of time.

[1] When Boethius (c480-524) in the mediaeval period came to translate Aristotle’s Organon (which included the book Categories) into Latin, he used the equivalent legal Latin term predication for the Greek category.

The intension-extension distinction is implicit in the Organon but not really made explicit until brought out in the Port Royal logic of 1662-1683 (From ideas 1662-1683 [1], Comprehension and Extension at p. 39-40):

> Now in these universal ideas there are two things which it is most important to distinguish clearly, *the comprehension* and *the extension*. I call the *comprehension* of an idea the attributes that it contains in itself, and that cannot be removed without destroying the idea. For example, the comprehension of the idea of a triangle contains extension, shape, three lines, three angles, and the equality of these three angles to two right angles, etc. I call the *extension* of an idea the subjects to which this idea applies. These are also called the inferiors of a general term, which is superior with respect to them. For example, the idea of a triangle in general extends to all the different species of triangles.

Sir Stanley Jevons (1832-1882) in his popular 19th century text on logic defines extension and intension thus:

> The extension, extent, breadth, denotation, domain, sphere or application of a name consists of the individual things to which the name *applies*. The intension, intent, depth, connotation, or implication of a name consists of the qualities the possession of which by those things is implied [2].

To explore therefore the typing of anticipatory systems we need to look more generally at the nature of type. It is a fundamental and was the great stimulus for philosophy originally for exploring the nature of the world. This ties up with logic as seen in Aristotle but until the 19th century logic was concerned only with categories until the semantics of Aristotelian syllogism was applied in the symbolic logic of Frege.

Alfred North Whitehead (1861-1947) and his student Bertrand Russell (1872-1970) attended the famous 1900 logic colloquium at the Sorbonne in Paris and were very impressed to hear Frege's exposition of his *Begriffsschrift* which they determined to develop. A watershed in the history of logic was this earlier publication [10] by Gottlob Frege (1848-1925) of his formula language, modelled upon that of arithmetic, for pure thought. *Begriffsschrift* is a word thought to have been coined [3] [4] of his 'ideography', a formula language for determining the validity of a sequence of logical consequence [preface] in translation to prevent anything intuitive [5].

[2] [15] at p.39 in Lesson V p.37-44. See also in Lesson VIII Kinds of Propositions at p.62.

[3] the word *Begriffsschrift* is found in a work of 1867 quoted in his preface (translation [10] by Heijenoort at p.1) by Frege for the 'conceptual content'.

[4] *Begrifflichen inhalt.*

[5] *Anschauliches.*

> From anything penetrating here unnoticed I had to bend every effort to keep the chain of inferences free from gaps. In attempting to comply with this requirement in the strictest possible way I found the inadequacy of language to be an obstacle; no matter how unwieldy the expressions I was willing to accept, I was less and less able, as the relations became more and more complex to attain the precision that my purpose required. This deficiency led me to the idea of the present ideography. Its first purpose therefore, is to provide us with the most reliable test of the validity of the chain of inferences and to point out every presupposition that tries to sneak in unnoticed, so that its origin can be investigated. That is to why I decided to forego expressing anything that is without significance for the inferential sequence. [10]

Whitehead and Russell both seemed to have earlier interest in the syntactico-semantic relationships but from different origins. Russell was taken with Meinong and viewed the topic as one of denotation with a well-received paper *On Denoting* [22] [6]. Whitehead on the other hand was rather taken with the work of Hermann Grassmann (1809-1877) [7]. Whitehead initiated universal algebra by his work of that title in 1898 [8].

3 Categories as Process Types

The arrow $\longrightarrow$ represents the essence of process at any level. It has a built-in order from tail to head, from start to finish or in standard category theory vocabulary from domain to codomain. Sometimes it is convenient to use the standard symbol $\leq$ for ordering but the convenience is more for historic reasons [9] than scientific specification because the equality part of $\leq$ is vague and category theory always requires equality to be precisely defined. Because the arrow is at any level it transcends the classical distinction between a sort and a type whereas a sort is like an alphabet and the type corresponds to the word i.e. a string or tuple constructed from the letters 'strung together'. In formal languages a word is normally a total ordering in syntax but in a natural language the ordering can be partial arising from the semantics embedded

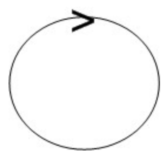
[6]This paper seems to still be of interest and was the subject of a recent conference: Russell v Meinong: 100 Years after On Denoting, McMaster University, Nicholas Griffin & Dale Jacquette, May 14-18 (2005). The authors are grateful to J. Chandler for drawing our attention to this.

[7]Self-taught mathematician but his mathematical work was not recognised in his own lifetime although as a philologist he had world renown for his work on Sanskrit.

[8]Whitehead laments that no one understood it [25]. However on the basis of it he was elected to a fellowship of the Royal Society.

[9]The less than or greater than part of this symbol (sometimes referred to as inequality) was first introduced by Thomas Harriot (1560-1621) under the patronage of the ninth earl of Northumberland [3]. The equals sign = itself conceived as two parallels was also popularised by Harriet although it was first introduced by Recorde [19].

in the ordering [10]. However this is more from the category than the predicate perspective. From the category/type perspective it is a pointer and represents either the verb 'sorts' or 'types' depending on the level. The arrow is not restricted like that for membership (ϵ) of a set, and can therefore be recursively applied, to give a generalisation of the fractal property. Note that the verbs 'sorts' and 'types' are proactive or functional in the sense that the arrow does not just indicate but may have the effect where necessary of transforming the object sort or type into that indicated in the arrow. The simplest is the arrow where the domain and codomain are indistinguishable. This is an arrow that points to itself:



The functional aspect is to self-identify and as an identity arrow may be defined as an object (from the set perspective) or a sort (from the process perspective). An arrow always does something even if it is just to identify itself.

This rough division into category and predicate of logic and language amounts to no more than an overlap of two viewpoints from very widely diverse perspectives. Since the arrival of category theory on the intellectual scene in the second half of the twentieth century many of the vague concepts can be put into the much sharper focus of formal definitions particularly with the discovery of universal principles of limits and colimits arising from adjointness. For historic reasons category theory has been developed within pure mathematics from a set theoretic basis. The category of sets however is still subject to classical limitations which are not part of the real world and are only applicable under very restricted conditions. Gödel (1906-1978) relying on the foundations of Whitehead and Russell's *Principia Mathematica* [26] showed that first-order predicate logic and therefore any interpretation or model was only complete and consistent if it was a first-order [11]. What is first-order is itself not too well defined. Russell in his development of ramified types defines higher-order propositions with variables that also contain variables. A variable is a symbol which is to have one of a certain set of values, without it being decided which one ([23]).

> Thus whatever contains an apparent variable must be of different type from the possible values of that variable; we will say that it is of a 'higher' type. Thus the apparent variables contained in an expression are what determines its type.

First-order therefore can cover quite a range of modelling by the use of clever mathematical methods which at times have more of the character of tricks and this

[10] perhaps first appreciated by Leibniz in his early essay *De Arte Combinatoria* [14].

accounts for the great success of the modelling technique through the last century including the development of quantum mechanics and the use by Einstein in his theory of relativity. Nevertheless a model to relate quantum mechanics and relativity within present knowledge would involve a variable relation between variables and therefore would be unlikely to satisfy Gödel's results [11]. Furthermore any representation in terms of number or sets and based on axioms is also undecidable. In the context here of categories and predicates a fundamental source of uncertainty is impredication as pointed out by André Poincaré [11].

Impredication arises when a predicate has greater scope than its subject to which it is applied as an attribute. Russell got round this difficulty in his ramified type theory by imposing the axiom of reducibility [23], sometimes known as an axiom of classes where a predicate is restricted to its subject. This is an arbitrary Procrustean action with unpredictable consequences when applied to the real world.

Around the time that Gödel was working on his theorems based on the work of Whitehead and Russell, Whitehead had moved away from classical models to the metaphysical concept of *process* [27]. It is this process version of category theory that escapes the problems raised by Gödel and Russell and to which we will turn in an attempt to clarify the predicative notion of *anticipation* whether as category or predicate.

Although the concept of limits and its dual the colimit has been apparent for some time it was not until the 1970s that their significance as universals became recognised. Mac Lane observes ([17] p.78):

> Universal arrows are unique only up to isomorphism; perhaps this lack of absolute uniqueness is why the notion was slow to develop. Examples had long been present; the bold step of really formulating the general notion of a universal arrow was taken by Samuel in 1948; the general notion was then lavishly popularised by Bourbaki.

In reality any composite object is a right-exact limit generated from its component co-exact colimits by process [20]. The colimits are a generalisation of the minima at any level of any theory, for example the elementary particles in the Standard Model of physical matter, atoms in classical chemistry, elements in set theory, etc. The colimits of process are conveniently represented by the arrow. The most fundamental arrow is as mentioned above the arrow that generates itself, the identity arrow that defines an object or a sort. The colimit between identity arrows then represents the generation of an object from another: $\grave{\mathrm{O}} \longrightarrow \grave{\mathrm{O}}$. Or the combination of a sort to form a type. It may readily be seen how these are all part of process. There will be a limit to all these 'up to natural isomorphism', that is the limit defined as a 'category' which is a type. Up to natural isomorphism therefore means that while different types are possible nevertheless types of types are indistinguishable.

[11] Poincaré's paper [18] is in three parts. The point on impredication is in the last of the three papers at p.307.

A category then gives a closure on type. This itself is an example of an important universal closure at the third interface ‘big arrow’, shown in Figure 1 as the lowest three interfacial closure identity arrow 1_l 'a sort itself'.

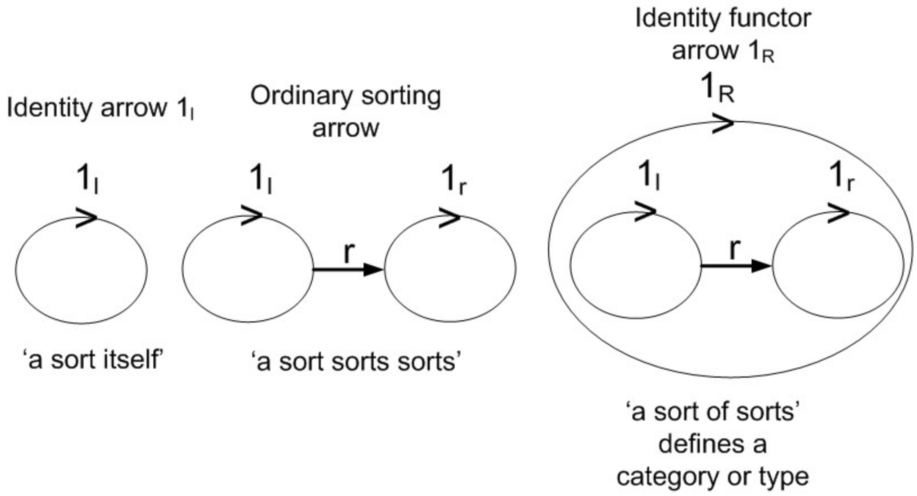


Fig. 1: The lowest three interfacial closure

The definition of a cartesian closed category is such that the identity functor is the terminal sort of the category with a unique distinguishable arrow from any sort in the category to the terminal sort. The consequences of this is that there is a unique composable arrow between any distinguishable sorts. The identity arrow has a contravariant dual generating arrow which is an initial sort with a uniquely distinguishable arrow from the initial sort to every sort in the category. The terminal sort is the absolute truth for that category although there may be relative local truths within the category. For example a and b may be distinguishable relative truths with respect to $e, i, o,$ and u as shown in Figure 2. Through composition, they are also relative truths with respect to y and z. The initial and terminal arrows, taken from Figure 1, are at the bottom and top of the diagram respectively. In a cartesian closed category no object can act as a generator for the category other than the initial object. Any other generating object has an arrow from the initial object to it. Whitehead and Russell pluck truth and falsehood out of the air; here we do it without any such tricks.

Most of the theoretical work to date on anticipatory systems relates to weak anticipation but the twenty first century has moved from local dynamical systems in physics where these methods may be quite adequate on to problems of globalisation and to very complex subject areas like biology and medicine which call for solutions with strong anticipation. As already implied, the troubles arise in the typing when there is a lack of tools powerful enough to produce results with strong anticipation. Twentieth century mathematics has been dominated by a logic which is Parmedian in extension as well as in its intension. The problems may be examined from the

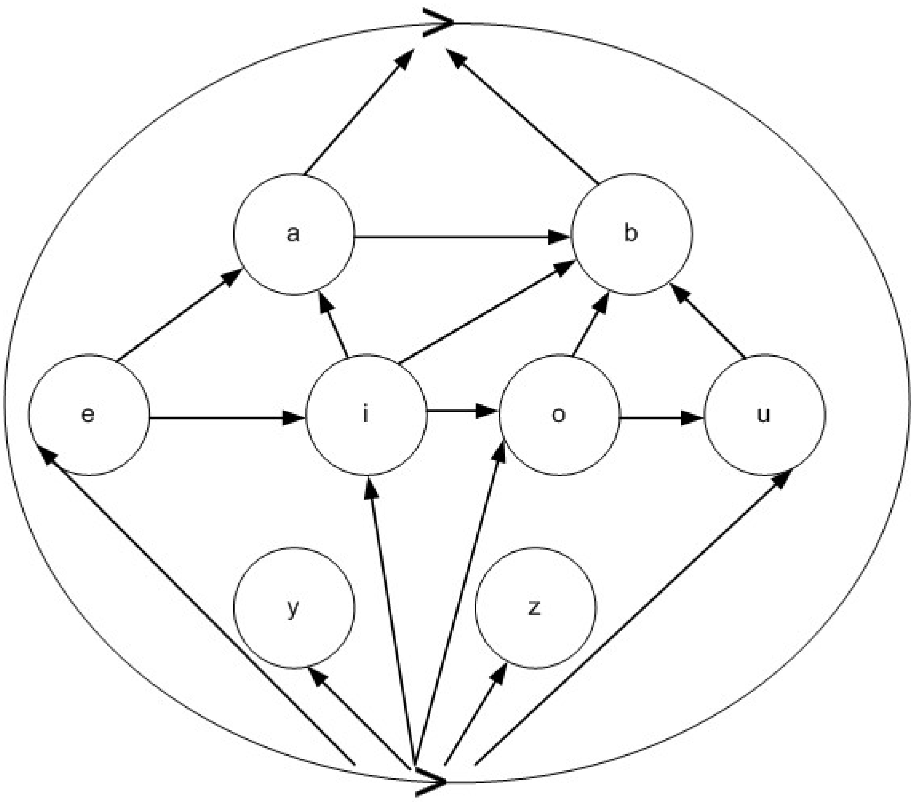


Fig. 2: Truth Types. Also initial arrows to all objects not shown beyond the first. Resultant by Composition

view points of the three eminent mathematicians.

- the undecidability of Gödel
- the paradox of Russell
- the impredication of Poincaré

Gödel has famously shown that both for intension and extension it is not possible to determine whether any system based on number and relying on axioms is true or false. This gives a general result that makes the goal of ultimate consistency within set theory unattainable. Russell's paradox and Poincaré's impredication are particular manifestations of Gödel's undecidability. As is well known Russell was acutely aware of the inadequacy of set membership because the set of all sets could not be a member of itself and from his study of denotational predication explored a number of advanced theories of typing to overcome the problems but on his own admission these did not succeed. Poincaré had already pointed out that the crux of the problem lies in the scoping of the predicate, which leads to three relevant strands:

- the logical system of Whitehead & Russell's Principia Mathematica [26] allows solely for a simple predicate giving rise only to weak anticipation
- a predicate has to be coextensive with its subject to give certainty, the axiom of extensibility or reducibility
- a predicate needs to be variable to allow for a varying context but even Frege and Russell could not agree on the meaning of *variable* [12].

4 Example of Data Typing

Real problems that arise in closed worlds of information systems provide very striking examples of the difficulties that result from simplification and normalisation of predicates. There is a demand for strong anticipation if information is to be reliably exchanged through open interoperable systems. This will only be realisable from the implementation of formal systems that can avoid the undecidability of Gödel.

Data typing has evolved considerably from the early days of computer science. Initially data typing was used as a constraint on the values that a particular variable could take. Such constraints were often also very broad such as integers, reals or strings, and often designed more to protect the computing system from run-time failures than to assist the user in securing more accurate storage of their data values. The relational model of Codd [4] provided an important conceptual advance with the development of domains, which in effect were pools of values that were available for instantiation by application programs. Moreover the domains would ideally be integrated across the entire system so that searches and views would all apply the same constraints. Domains thus enable a refinement of the basic type system to include restrictions on the range of values that are available. Basic data types such as integer, real and string had restrictions on the operations available. For instance: integers and reals might be multiplied but not strings; reals rounded but not integers. It was soon realised that the same operator could be used for different types but with a subtle change in meaning. Adding two integers is arithmetical addition, adding two strings together is concatenation. Ullman ([24] p.22-23) did not deal with type as an important characteristic of data models, preferring to dwell in the context of knowledge on aspects such as declarativeness. He considered that data models were either value-oriented, such as the relational, or object-oriented. With respect to the latter complex data types would be constructed from primitive data types

[12] Frege himself questioned Russells definition on the word variable and the significance of 'a symbol has a value' and suggested that Russell's definition should be rephrased as a variable is one of a certain set of values, without it being decided which one' (translation [10] by Heijenoort at p.10).

such as integer. Interestingly Ullman thought that object-identity preservation and encapsulation were antithetical to declarativeness.

In the 1980s a further significant development was with the abstract data type (ADT). This extended the domain concept by restricting the operations that could be performed to those specified by the creator of the type. For instance it might be decided that for a data type handling weights, that addition, subtraction and multiplication are valid operations but that multiplication is not. In ADT the functionality of the date type is restricted by providing an interface of allowable functions to restrict the use of the type. The ADT though was not just concerned with restrictions: it is also simple to provide more complex functions tailored to a particular task which might be quite complex for end-users to achieve on their own. Such ideas became part of the object-oriented paradigm where the functions were called methods and further extensions were made to the concept of the data type. Some motivation for the new developments had already been mooted in work on data abstractions where abstractions such as inheritance and aggregation had been identified [16]. Parallel work by Codd on a new relational model RM/T [5] had also identified similar abstractions within a directed-graph framework.

The object-oriented paradigm adopted as a core principle the ideas of data subtypes for inheritance whereby a subtype would inherit all the properties of a parent supertype (attributes, constraints, methods) but would also have its own attributes and methods as a specialisation of the supertype. Types could thus be held as a hierarchy (single inheritance) or a network (multiple inheritance). Aggregation is achieved by bringing together types into complex groupings, for instance assembling the data types for the components of a car into a new data type for the whole car. Other abstractions introduced were generalisation, in the opposite direction to specialisation.

While the object-oriented approach appears to offer richer typing, Codd said in 1990 ([6] at p.22):

> One of the main reasons that object-oriented DBMS and prototype products are not going to replace the relational model and associated DBMS products is their systems appear to omit support for predicate logic. It will take brilliant logicians to invent a tool as powerful as predicate logic. Even then such an invention is not an overnight task. Once invented it may take more than a decade to be accepted by logicians. Thus features that capture more of the meaning of data, which is important, should be added to the relational model, instead of being proposed as replacements.

Date & Darwen in their Third Manifesto ([7] at p.421) assert that names are values, so all names exist *a priori*. Sets are values, so all sets exist *a priori*. Types are <name, set> pairs, so all types exist *a priori*. This maintains a general set-based approach. However, it ignores behaviour. The top level in a typing system is the transformation of types with process as the ultimate closure.

5 Monad/Comonad as Anticipatory System

A monad is sometimes described as a triple, comprising an endofunctor say T, the unit of the monad η and the multiplication of the monad $\mu : T^2 \longrightarrow T$. The monad is traditionally represented in terms of three parameters [13]:

$$< T, \eta, \mu > \tag{1}$$

A pair of adjoint functors is an endofunctor: in this case the source category of F, $\mathbf{L}$, is also the target category of G. So for the endofunctor T as the pair of adjoint functors GF, $F : \mathbf{L} \longrightarrow \mathbf{R}$ and $G : \mathbf{R} \longrightarrow \mathbf{L}$, the monad is instantiated as:

$$< GF, \mathbf{1_L} \longrightarrow GF, GFGF \longrightarrow GF > \tag{2}$$

where $\mathbf{1_L} \longrightarrow GF$ is the unit (η) of the monad and $GFGF \longrightarrow GF$ is the multiplication (μ).

The monad gives the left-hand perspective. There is also a dual comonad which gives the right-hand perspective, a perspective that is of significance for anticipation. A comonad is a 'triple', comprising S an endofunctor, ϵ the counit of adjunction of the comonad and δ the comultiplication of the comonad given by $\delta : S \longrightarrow S^2$. The corresponding 'triple' for the comonad is:

$$< S, \epsilon, \delta > \tag{3}$$

A pair of adjoint functors is an endofunctor: in this case the source category of G, $\mathbf{R}$, is also the target category of F. So for the endofunctor S as the pair of adjoint functors FG, $G : \mathbf{R} \longrightarrow \mathbf{L}$ and $F : \mathbf{L} \longrightarrow \mathbf{R}$. The comonad is instantiated as:

$$< FG, FG \longrightarrow \mathbf{1_R}, FG \longrightarrow FGFG > \tag{4}$$

where $FG \longrightarrow \mathbf{1_R}$ is the counit (ϵ) of adjunction of the comonad and $FG \longrightarrow FGFG$ the comultiplication (δ).

Further technical details of the monad and comonad are given in our companion paper on time jitter, including the adjointness between the monad and comonad [13] as confirmed by Barr & Wells ([2] at pp. 136-137).

Subsequent states of the extension of an anticipatory system evolve by a unitary emergent transformation of the intension. The monad and comonad provide two distinct formal views of emergent and evolutionary adjointness between the intension

[13] Some authors such as Barr and Wells [2] use the term *triple* for the monad. While the three parameters may be given as a tuple i.e. an ordered set, it should be noted that the three parameters are not independent elements for the purposes of set theory. This is perhaps an overloading of the word tuple.

and the extensions. The measure of emergence is given by the natural transformation, that is the multiplication unit of the emergence $\mu : T^2 \longrightarrow T$ while the measure of the dual evolution is given by the comultiplication unit of evolution $\delta : S \longrightarrow S^2$. The unitary transformation T is synonymous with the process we call the anticipatory system. T^2 is the transformation of the transformation $\mu : T^2 \longrightarrow T$ and therefore is a typing of the system T from its subsequent state T^2. This is a formal definition for anticipation. In language terms the predictive is the predicate of the system. The future is in the structure of the extension.

The dual measure in the comonad $\delta : S \longrightarrow S^2$ is "the present typing the future". As an operation comultiplication is addition so the system evolves by an additive process. Anticipation is the multiplication of the monad $\mu : T^2 \longrightarrow T$, where the future system types the present. Its dual is co-anticipation, the co-multiplication (i.e. addition) $\delta : S \longrightarrow S^2$, where history repeats itself. When $\delta = 1$ then $1 + 1 = 1$ and the system adds itself to itself i.e. history repeats itself. At lesser values of δ, history repeats itself to a lesser extent. It looks then that the concept of anticipation needs both the monad and the comonad as induced by adjointness which should be expected.

Anticipation and co-anticipation therefore transcends time. It is important not to be carried away with some mystical view of time but to recall that there are many subtypes of time, including sidereal time, solar time, atomic time, dynamical time, proper time and coordinate time, as well as variants on them [13]. Time belongs to the extension not intension of a system [21]. In everyday experience we identify time with some particular linear sequence of events. The structural arrangement of the events is a partial order as in Figure 2 [14]. We take it to be a particular linear thread through this partial order in our recognition of time.

6 Significance for General Systems

Because we are dealing with anticipatory systems in an underlying physical universe, all systems are only as defined in a cartesian closed category and one that is locally cartesian closed [2, 21].

What we have described in relation to anticipation is in terms of the properties of adjointness without applying any special assumptions or qualifications. It would therefore seem to follow that not only is the property of anticipation and co-anticipation a general one but all systems in the universe are anticipatory systems. The other interesting result is in the significance of the nature of time. Anticipation is more fundamental than time. Anticipation from its origins in the structure of adjointness can provide a full rigorous definition of time itself.

[14]Possible partial orders are quotient extension values of the preorder in the intension.

References

[1] Arnauld, Antoine, & Nicole, Pierre, *Logic or the Art of Thinking*, Translated Buroker, Jill Vance, Cambridge (1996).

[2] Barr, M, & Wells, C, Toposes, Triples and Theories, Springer-Verlag, New York (1985).

[3] Cajori, Florian, *A History of Mathematical Notation*, 2 volumes, Chicago **I** 199-201, **II** 115-117 (1928).

[4] Codd, E F, A Relational Model of Data for Large Shared Data Banks, *CACM* **13**(6) 377-387 (1970).

[5] Codd, E F, Extending the Database Relational Model to Capture More Meaning, *ACM TODS*, **4**(4) 397-434 (1979).

[6] Codd, E F, *The Relational Model for Database Management* Addison-Wesley (1990).

[7] Date, C J, & Darwen, Hugh,*Databases, Types, and the Relational Model: The Third Manifesto*, 3rd edition, Addison-Wesley, Boston (2006).

[8] Dubois, D M, Review of Incursive, Hyperincursive and Anticipatory Systems - Foundation of Anticipation in Electromagnetism. *CASYS'99 - Third Int Conf*, Dubois, D M, (ed), AIP Conference Proceedings **517** 3-30 (2000).

[9] Dubois, Daniel M, Mathematical Foundations of Discrete and Functional Systems with Strong and Weak Anticipations, in: Anticipatory Behavior in Adaptive Learning Systems, State-of-the-Art Survey, Edited by Martin Butz et al, Lecture Notes in Artificial Intelligence, Springer **2684** 110-132 (2003).

[10] Frege, Gottlob, (1879), Begriffsschrift, eine der arithmatischen nachgebildete Formelsprache des reinen Denkens (Halle). Translated in Jean van Heijenoort, *From Frege to Gödel: A Source Book in Mathematical Logic 1879-1931* Harvard 1-82 (1967); also reprinted in Angelelli, Ignacio, Begriffsschrift und andere Aufsätze, Olms, Hildesheim (1964).

[11] Gödel, Kurt, (1931), Über ber formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme, I. *Monatshefte für Mathematik und Physik* **38** 173-98. Translated in Jean van Heijenoort, *From Frege to Gödel: A Source Book on Mathematical Logic* Harvard 596-616 (1967).

[12] Heather, M A, & Rossiter, B N, Locality, Weak or Strong Anticipation and Quantum Computing I. Non-locality in Quantum Theory, *International Journal Computing Anticipatory Systems* **13** 307-326 (2002).

[13] Heather, Michael, Rossiter, Nick, & Sisiaridis, Dimitris, The Semantics of Jitter in Anticipating Time Itself within Nano-Technology, CASYS'07, Lige, AIP Conference Proceedings 12pp (2008).

[14] Leibniz, G W, Dissertatio de arte combinatoria, *Sămtliche Schriften und Briefe*, A VI 1, p. 163, Akademie Verlag, Berlin 1923- (1666).

[15] Jevons, W Stanley, Elementary Lessons in Logic, Macmillan (1870).

[16] Lockemann, Peter C, Mayr, Heinrich C, Weil, Wolfgang H, & Wohlleber, Wolfgang H, Data abstractions for database systems, *ACM TODS*, **4**(1)60-75 (1979).

[17] Mac Lane, *Categories for the Working Mathematician*, 2nd ed, Springer (1998).

[18] Poincaré, Les Mathématiques Logiques, *Revue de Métaphysique*, **13** 815-835 (1905), **14** 17-34, *ibid* 294-317 (1906).

[19] Recorde, Robert, *Whetstone of Witte* (1557).

[20] Rossiter, B N, Heather, M A, & Sisiaridis, D, Process as a World Transaction, *Proceedings ANPA 27 Conceptions*, 122-157 (2006).

[21] Rossiter, N, & Heather, M A, What are Bit Strings? The View from Process, ANPA, Cambridge (2008).

[22] Russell, Bertrand, On Denoting, *Mind* **14** 479-493 (1905).

[23] Russell, Bertrand, Mathematical Logic as based on the Theory of Types, *American Journal of Mathematics* **30** 222-262 (1908).

[24] Ullman, Jeffrey D, Principles of Database and Knowledge-base Systems **1**, Computer Science Press, Maryland (1988).

[25] Whitehead, Alfred North, *A Treatise on Universal Algebra*, Cambridge. (1898).

[26] Whitehead, A N, & Russell, B, *Principia Mathematica* 3 volumes, Cambridge, 2nd ed (1925-27).

[27] Whitehead, Alfred North, *Process and Reality: An Essay in Cosmology*, Macmillan, New York (1929); corrected edition, edd D R Griffin, D W. Sherburne, New York, The Free Press (1978).