INTENSIONAL LOGIC IN CONTEXT

WILLIAM W. WADGE
Department of Computer Science, University of Victoria,
P.O. Box 1700, Victoria, B.C., Canada V8W 3P6
E-mail: wwadge@csc.uvic.ca

What exactly is "Intensional programming"? Every year, at every *ISLIP*, we try to answer this question. And the answers keep getting better. There is, of course, an easy answer, namely "programming in a language based on intensional logic". But this answer raises another, more fundamental question: "What is intensional logic?" Logicians have been trying to answer that question for about 2500 years, and their answers are also getting better. In this tutorial I will present some of these answers to the second question and explain how they help answer the first.

1 Intension vs Extension

The term "intensional" itself is relatively recent — Carnap introduced it in the 1930s, based on Frege's distinction between the "sense" and the "denotation" of an expression. In Frege's terminology the denotation of an expression is just that — the particular object it in fact denotes (or currently denotes). (This is what Carnap and most modern logicians call the extension). On the other hand, the "sense" (what we now call the intension) is the entire concept it represents — what we, at some level, intend when we write it. For example, the expression "the President of France" denotes currently Jacques Chirac; but no one would claim that M Chirac somehow sums up the whole concept of the French presidency.

Intensional logic is therefore the logic of expressions in which the intension of subexpressions (and not just their extensions) have to be taken into account. These are very common in natural language. The French constitution specifies that the French President is directly elected; and this is not the same as specifying that M Chirac be directly elected.

2 The mysteries of intensionality

Many famous paradoxes are based on the observation that intensional expressions seem to violate the basic law of substitution of equals for equals. For example, it is certainly true that

The number of planets = 9

Furthermore Kepler, the famous astronomer, was well aware of the basic rules of arithmetic; we can be sure that

Kepler knew that 9 is a perfect square

But if we substitute equals for equals in this latter assertion, we derive

Kepler knew that the number of planets is a perfect square

which is almost certainly false.

In one sense, the explanation is simple: the equation relates only the extensions of the two expressions, whereas the assertion "Kepler knew ..." refers to the intension "the number of planets". But what sort of object is an intension?

Aristotle, the founder of formal logic, first addressed similar problems. It is often said that Aristotle's logic is strictly two-valued, but this is not correct. He carefully distinguished between assertions that are true but not *necessarily* so, and those that are true by necessity — that could not possibly be false. Aristotle classified these different "modes" of truth and falsity. He extended his analysis of syllogisms to include those in which assumptions and/or conclusions were not simply true, but necessarily true, or only possibly true.

Necessity, therefore, is in our terminology an intensional operator. We cannot completely determine the truth of "Necessarily P" knowing only the truth (extension) of P. P may be true, but not necessarily so.

The Greek Stoics and the medieval Scholastics continued this tradition. During this entire period of more than two millennia, "modal" logic was considered to be an integral part of formal logic.

3 The mechanization of logic

The split between modal and "conventional" logic began with Leibniz' program to mechanize mathematics (including logic) and turn it into a form of calculus. Boole's discovery of the arithmetic of truth-values seems to exclude modal logic from this program. Even if we view the assertion

It is possible that he will arrive tonight

as somehow being the result of applying a possibility operator to the assertion

He will arrive tonight

what could its truth table be? If A is true, it clearly must be possible that A be true. But if A is false — happens to be false — we cannot conclude that it is impossible.

This phenomenon is simply the Boolean version of the paradox cited above. If A is false, then (in Boolean logic) it is equivalent — equal — to any other false assertion, e.g. $1 + 1 = 3$. But we cannot conclude that the possibility of A is equivalent to the possibility of $1 + 1 = 3$ being true.

If we think a bit about the nature of possibility and necessity we can see why they would not fit easily into the mechanical/calculational model. Suppose that the students in a class take a midterm and are surprised to see that there are no questions on chapter one. The factual question, whether or not there are any chapter one questions, can be settled easily enough, simply by examining the book and the midterm. But, should the students be surprised? If their surprise was warranted, it follows that the omission was not necessary; that it was possible that there would be chapter one questions, and it merely happened that there were none.

However, the necessity or otherwise of the omission cannot be settled by referring only to that particular midterm in that particular course in that particular semester, etc. It depends on the entire context in which the event (the midterm examination) took place.

The mechanical/calculational model is based on the principle that the whole is the sum of its parts. In logic, this is called referential transparency. It is based on the assumption that the parts are assembled in a vacuum, and that there is no context to take into account. This is in fact a very modern concept.

4 Logic in a social context

In the ancient world, social life was never as fully pigeonholed/atomized as it is today. Aristotle, for example, was not a "logician" or a "biologist" or a "political scientist"; he studied every aspect of the world, as did his teachers, his pupils and his colleagues.

Nevertheless, we can understand why it was first in the city-states of classical Greece that logic and abstract science in general emerged. They were the first societies in which money and the marketplace (the agora) played an important part. In particular, in the agora the ancient Greeks could experience firsthand the phenomenon of evaluation. A carpet (say) is brought to the agora and exchanged for a sum of money — only as much as the buyers are willing to pay: its *value*. The carpet is taken out of the house or workshop where it was made, separated from those who made it and possibly used it. In the marketplace, the context from which came the carpet is irrelevant. The carpet moves from buyer to seller unimpeded by any attachments — in a vacuum.

It is not a great leap to apply the same evaluation process to speech itself — to distinguish the truth *value* of an assertion from everything else about it. In fact, many of the classical philosophers had no choice in the matter, since they taught (spoke) — like Aristotle — for a fee.

Modern, extensional logic is really the logic of events that take place in a vacuum — the logic of empty space. It corresponds closely to Euclidean geometry, which is the geometry of space that is flat (empty of features). In fact, according to general relativity, Euclidean geometry fails in the real world precisely because space is not empty (of matter and gravitational fields).

Aristotle was in fact highly critical of the notion of empty space and in general of the theories of the early atomists. (The atomic theory of Democritus described a universe of dense atomic particles moving about in a perfect void). This debate, about the possibility of a perfect vacuum, continued for more than two millennia until the beginning of the twentieth century, when Einstein's relativity theory apparently abolished the electromagnetic aether.

5 The fall and rise of modal logic

At almost the same time that Einstein's special relativity abolished the aether and declared space (i.e. the space between atoms) to be empty , Frege and Russell completed Leibniz' program of mechanizing logic. Set theory and the predicate calculus are entirely extensional formalisms and deal with unchanging, immortal entities existing and encountering each other in an empty context. Extensional logic proved very successful, and Russell and Wittgenstein wasted no time in generalizing their approach to a whole philosophy, *logical atomism*. In this philosophy knowledge — in fact reality itself — can be described as a large collection of atomic facts, each of which can be evaluated in isolation.

Frege was aware that something (namely, intensions) had been overlooked, but was unable to incorporate them in his system. Efforts began immediately to extend the mathematical logic to cover intensional phenomena (as they were later called).

At first, the problems with substitutivity explained above seemed to block any formal semantics of intensionality. Instead, logicians took a purely syntactic approach.

The effort to recover modal logic began soon after, when C. I. Lewis introduced his logic of *material implication*. Lewis avoided the semantic problems associated with the vague concept of an "intension" and instead took a syntactic/proof-theoretic approach. He began with (extensional) propositional logic and added another connective, $\prec$, which he called strict implication. His motivation was to avoid the paradoxes associated with ordinary, extensional ("material") implication as defined by the familiar truth table. Lewis gave a number of axioms for the extended system, axioms that included for $\prec$ only some of those whose analogs are true of $\rightarrow$.

Lewis and his collaborators soon realized that it was more convenient to use as primitives the classical notions of "necessity" and "possibility", as additional unary operators. For "possibility" Lewis introduced the symbol $\Diamond$, and for necessity the symbol $\Box$ (these have since become standard). Strict implication became a derived operator: $(P \prec Q)$ was defined as $\neg\Diamond(P \wedge \neg Q)$. In other words P strictly implies Q iff it is not possible that P be true and Q be false simultaneously.

In 1932 Lewis and Langford published the first systematic treatment of modal logic. One of the challenges was that it is not entirely obvious which axioms are needed to capture the notions of necessity and possibility. For example, Aristotle noted that everything that is necessary is possible, i.e.

$$\Box P \rightarrow \Diamond P$$

More generally, everything that is necessary is in fact true; and everything that is in fact true must clearly be possible. Hence

$$\Box P \rightarrow P$$

and

$$P \rightarrow \Diamond P$$

On the other hand, if P and Q are both possible, we cannot assume that they can be true simultaneously; so

$$(\Diamond P \wedge \Diamond Q) \rightarrow \Diamond(P \wedge Q)$$

cannot be an axiom or even a theorem.

Some principles, however, are hard to decide. If something is possible, is it necessary that it be so? In other words, should

$$\Diamond P \rightarrow \Box\Diamond P$$

be a tautology? Lewis and Langford avoided ruling on such contentious issues and instead developed a series (S1 through S5) of (propositional) modal logic systems based on increasingly more powerful axioms. Since then logicians have proposed literally dozens of different theories, and the whole collection is far from being linearly ordered.

6 The formalization of intensional logic

About the same time Rudolf Carnap formalized the notion of intensional and extensional contexts. For example, in

Kepler knew that 9 is a perfect square

the name "Kepler" appears in an extensional context — we can replace it with another expression denoting the same individual, without affecting the truth value of the statement. The numeral "9", however appears in an intensional context and cannot be replaced, as we have already seen. In natural language, verbs such as "know", "believe", "wish" and so on introduce intensional contexts. The same is true of impersonal qualifiers such as "must", "ought", "may", "can" and the like. And, of course, "necessity" and "possibility". Carnap distinguished between extensional and intensional equivalence and proposed the principle that intensionally equivalent expressions can be substituted even in intensional contexts.

The syntactic approach confirmed that modal logic is not inherent nonsense; Lewis and others who followed showed that their systems are consistent, and do not collapse the modal operators onto the identity operator. Nevertheless, their system had no model theory (no *semantics* in computer science terminology). As a result, for at least another three decades modal logic lacked the respectability of extensional logic.

The lack of model theory created at least one serious philosophical problem, namely what are the objects of (for example) our beliefs? The purely syntactic approach leads to the conclusion that we believe in syntax. In other words

I know that the number of planets is 9

asserts a relation between me and the phrase (string) "the number of planets". The paradox is now trivially explained: substitution changes this string, as a string, possibly to a string that is not an object of my beliefs. (Church proposed a more sophisticated version, in which the string is replaced with a combinator and arguments.)

The purely syntactic approach has a certain kind of philosophical self-consistency. Also, it was very much in tune with the logical positivist school of the 1930's. However, it proved useless for producing new technology — new logical tools for analyzing and reproducing intensional phenomena. Logical positivism passed out of favor and the search for a semantics of intensionality continued.

(We should mention that the purely syntactic approach is far from dead. To many people, an "intensional approach" means an approach based on manipulating expressions rather than the entities they denote.)

7 Kripke's possible worlds semantics for modal logic

Finally in the early sixties Saul Kripke presented the first completely formal semantics for modal logic. This semantics was based on the notion of "possible world" or "alternate state of affairs", which can be traced back nearly a thousand years, to the medieval logician Duns Scotus. For Scotus, an event is possible means that we can imagine it taking place without contradiction in at least one alternate state of affairs. Conversely, something is necessary if it remains true in all the alternate states of affairs.

The problem with using possible worlds for modal semantics is that on the face of it, a possible world is a rather complex entity. Originally, it was a complete, alternate state of affairs — a parallel world or universe in which some things are as they are in this world (the "actual" world), and some things are different. (Parallel universes are a common theme in contemporary science fiction.) The whole concept is philosophically suspicious. If everything happens for a reason (has a cause), how can anything be different?

Kripke's contribution (anticipated by Church and Carnap) was to completely avoid the whole question and to take "possible world" to be an undefined concept. Every "Kripke model" has a set of possible worlds but they have no real structure; they are mathematical black dots, used essentially as indices. A Kripke model is simply an indexed family of normal interpretations, one for each possible world. The only structure on the universe of possible worlds is a binary relation that specifies which worlds are possible alternatives to a single given world (this is called the *accessibility* relation).

In Kripke semantics, propositions are not normally either true or false; truth is relative to a particular world, and so propositions may be true in (at) some worlds and false in others. The Kripke semantics of necessity and possibility formalizes the "alternate state of affairs" idea of Scotus. A proposition of the form $\Box P$ is true at a world w iff P is true at *some* worlds w' accessible from w. Dually, a proposition of the form $\Diamond P$ is true at a world w iff P is true at *all* worlds w' accessible from w.

Kripke showed that most of the proposed modal axioms corresponded to different assumptions about the accessibility ordering. For example, if everything necessary is to be true, and everything true is to be possible, we require that the relation be reflexive (so that every world is an alternate to itself). If the ordering is transitive, we have

$$\Box P \rightarrow \Box\Box P$$

so that everything necessary is necessarily so.

Kripke models were a turning point for the development of intensional logic. The subject could no longer be dismissed as a meaningless formalism inspired by the supposedly undisciplined illogicalities of natural language.

Kripke models also had the effect of demystifying the concept of possible world. They showed that one could use the concept without having a complete definition of the notion (in much the same way that propositional logic allows us to manipulate propositions without committing ourselves to a particular view of what constitutes a "proposition").

In fact, in many applications the so-called "possible worlds" are simple entities, better thought of as coordinates or indices. In particular (as Prior was first to point out), we can take the "worlds" to be timepoints, and make every point accessible to every other. The result is a simple temporal logic, with $\Box$ meaning *always* and $\Diamond$ meaning *sometimes*. Better yet, we can specify that only points in the future are accessible from a given point. With this relation $\Box$ means *henceforth* (at all times in the future) and $\Diamond$ means *eventually* (at some time in the future).

8 Scott's semantics for intensional logic

Once the notion of possible world/context was formalized, it was easy to find new intensional formalisms that went beyond the traditional necessity/possibility. For one thing, there is no need to restrict ourselves to a single pair of modal operators. In temporal logic, we can choose to make $\Box$ and $\Diamond$ look forward, as above, but add another pair based on a different relation in which past timepoints only are accessible from the present. In addition, if time is discrete and only the next point is accessible, both $\Box$ and $\Diamond$ mean *next*.

The most important generalization, however, was to break any remaining ties with the ancient modalities and consider operators not defined by an accessibility relation. This means giving a semantics not just to modal logic, but to intensional logic in general. Credit for this primarily goes to Dana Scott. In 1969 he laid out a complete framework for a semantics of intensional logic in his (perhaps mistitled) *Advice on Modal Logic*.

This remarkable paper is so clear and prescient that anyone reading this tutorial should read it as well. However, for the sake of completeness we will summarize the main points. In *Advice* Scott takes the basic idea of Kripke models and extends it to give a framework in which Carnap's distinction between intension and extension can be formalized. In what should be called a "Scott model" (or a Scott-Montague model) we have a nonempty set I of reference points (essentially, possible worlds) but do not require that any accessibility relations be specified. Whatever the syntactic details of our language, propositions are not a priori absolutely true or false (as is the case with Kripke models). Their truth value varies from world to world; although they may happen to have the same value at each world.

Scott calls the truth value of ϕ at a particular world the *extension* of ϕ at that world. The *intension* of ϕ, on the other hand, he takes to be function that maps each world w to the extension of ϕ at w. In other words, the intension of a formula is an element of 2^I, 2 being the set $\{0,1\}$ of truth values. In Scott's approach, a (unary) intensional operator is simply a function that maps intensions to intensions, i.e. an element of $2^I \to 2^I$. These functions may be defined in terms of an accessibility relation, but the framework allows for more general kinds of operators.

As an example, he proposes a formalization of the present progressive tense (as in "I am eating"). He takes I to be the reals and defines $[\leftrightarrow]E$ to be true at time t iff E is true in some interval (however small) containing t.

Scott models are not restricted to propositional logics. They can also specify a collection D of individuals that serve as the extensions of terms. These individuals can be (understood to be) numbers, strings and lists, physical objects, people, organizations, etc. However intensional individuals denote elements of D^I — they may have different extensions at different worlds. The elements of D^I are *intensional objects* or *virtual individuals.* They correspond to natural language phrases such as "the President of France" and Scott models can give us a clear explanation of the puzzles described at the beginning of this article. For example, the typical news story about the French President refers to the current extension of this virtual individual. (In other words, the term "the President of France" occurs in an extensional context). The French Constitution, however is about the entire intension of the phrase — about everything he/she might be, not just about what he happens to be here and now. In the Constitution the phrase is used in an intensional context.

(Actually the problem of individuals in intensional logic is more complex than indicated here. Scott devotes most of *Advice* to dealing with existence (some individuals may not exist in some worlds) and coincidence (different individuals may be indistinguishable in certain worlds). Up to now, Intensional Programming has found the simpler framework to be sufficient.)

The Scott (–Montague–Kripke–Carnap–Lewis) approach to modal and intensional logic has proved very successful in illuminating the foundations of logic, in understanding the complexity of natural language, and in representing and generating digital knowledge. Intensional programming is an effort to apply these concepts even more directly, as the basis of general purpose programming systems and models of computation.

9 From intensional logic to intensional programming

This is not the place to present even a brief overview of work in intensional programming. However, we can explain the connection by presenting a Creation Myth that makes the connection clear (and also puts its creators in a more favorable light).

According to the Myth, far-sighted Computer Science researchers read and understood *Advice* and decided to use it prescriptively, as the basis for new programming languages and systems. The first language was Lucid, invented by the author and E. A. Ashcroft in 1974. (Now you know who the far-sighted researchers were. However the Myth does not explain why they waited five years.)

The Creators chose (a family of) Scott models in which I is the set of natural numbers, interpreted as timepoints. For the individuals, they took (in the simplest case) D to be the set of integers. Their language included the operators `first`, `next` and `fby`, with `next` (for example) denoting the intensional operator `next` from $D^I \to D^I$ such that

$$\texttt{next}(X) = \lambda n.X(n+1)$$

Programs consisted of equations defining program variables in terms of constants and each other. Program variables are therefore intensional integers. It was an easy matter to allow programmers to define their own functions, that were not restricted to using their arguments in extensional contexts.

Finally, the Creators realized the language could be implemented through a demand-driven dataflow model, in which demands for particular extensions of variables can generate demands for other particular extensions.

10 Go forth and multiply — your dimensions

At this point the Myth has the Creators look at what they had done, and see that it was Good (a few referee reports aside). They return for more *Advice* and find it; for example,

> *This situation is easily appreciated where I is the context of time-dependent statements; that is in the case where I represents the instants of time. For more general situations one must not think of*

the $i \in I$ as anything as simple as instants of time or even possible worlds. In general we will have

$$i = (w, t, p, a)$$

where the index i has coordinates; for example w is a world, t is a time, $p = (x, y, z)$ is a (3-dimensional) position in the world, a is an agent [this is in 1969!], *etc. All these coordinates can be varied, possibly independently, and thus affect the truth of statements which have indirect references to these coordinates.*

The newly inspired Creators (by now a much larger group) first added extra time dimensions to Lucid to allow nested iterations. Then space dimensions, used much like arrays. The "place" dimension (actually a form of branching time) allowed programs with first-order recursive functions to be translated into pure intensional programs.

Later, spreadsheet and attribute grammar tools used cell and tree node coordinates, respectively. Branching time allowed logic programs to express search strategies in a very simple way. We also discovered a translation scheme for some higher-order programs using multiple-place coordinates.

11 Myth meets reality

Intensional programming obviously developed quickly and rationally in the parallel universe of the mythical Creators. Over in the real world, however, the real creation was a longer and less orderly process. To be fair to the real creators, they had some real obstacles to overcome, and in so doing they had to extend the system of *Advice* in important ways.

From the beginning they — we — wanted to allow recursive definitions of programs. This meant that the collection of intensions had to form a domain (yes, a Scott domain) and that the basic operators had to be continuous over this domain. In particular, this meant that the operators be finitary: that any particular extension of the result of an operation can be computed from a finite set of extensions of the operands.

Intensional programming also used multiple dimensions almost from the beginning. It took a long time, however to realize that it was not enough to have (even a large) set of fixed predefined dimensions; or even an infinite, indexed collection of predefined dimensions. Eventually, the GLU system allowed user-declarable local dimensions. More recently, work on versioning is based on a system with an unlimited supply of dimension identifiers; and in many applications, dozens of them are actually used.

In the end, the real creators realized that intensional programming systems need not be based on a single language or even a single paradigm. We (and by then there were a lot of us) realized that the basic notions of intensional logic — possible worlds and intensional operators — transcend any particular view of computation and are applicable in almost any context. At this point the Myth merged with reality (this happened about the time of the 1995 *ISLIP*, as presented in the first *Intensional Programming* volume).

Since then we have extended the *Advice* framework in two (more) important ways. First, with intensional versioning we introduced a partial refinement order to allow inheritance between possible worlds. Secondly, we allow dimensions to be manipulated as extensions — to be used as operands/arguments of expressions and functions, and to be returned as their values.

12 From philosophy to technology

The system presented in *Advice* has proved to be almost exactly that needed as a foundation for intensional programming; we have made no changes, and very few additions. It has passed the hardest design test of all, namely suitability for a purpose (intensional programming) that could not be foreseen.

The same, in fact, can be said for intensional logic as a whole. It has developed in much the same way as other, very different technologies. It began with the study of a few curious phenomena and a collection of baffling paradoxes. A long period of study and experimentation, much of it apparently futile, eventually led to understanding and to the perfection of useful tools. We have now reached the stage where, in Scott's words

> *... the old puzzles can be cast aside, and one can begin to provide meaningful applications.*

13 Short annotated bibliography

I. M. Bochenski. *A History of Formal Logic*. Translated and edited by Ivo Thomas. Chelsea Publishing Company, New York, 1956. 567pp.

> *An excellent history, from the days of Aristotle through the 1950s. Treats the ancient and medieval logicians with the respect they deserve, and gives careful consideration to modal logic.*

G. E. Hughes and M. J. Cresswell. *An Introduction to Modal Logic*. Methuen and Company, London, 1968. 388pp.

The standard reference work on modal logic. Thorough, but lacks motivation and does not address more general intensional logics.

Marshall McLuhan. *The Gutenberg Galaxy.* University of Toronto Press, Toronto, 1962. 294pp.

Amazing insights into the social and intellectual changes that began with the introduction of alphabetic writing and accelerated with the advent of the printing press.

Alfred Sohn-Rethel. *Intellectual and Manual Labor: A Critique of Epistemology.* Macmillan, London, 1978. 216pp.

Argues persuasively that money and the market economy layed the basis for scientific abstraction.

Dana Scott. Advice on modal logic. In Karel Lambert, ed., *Philosophical Problems in Logic.* D. Reidel, Dordrecht (Holland), 1970. 176pp.

Good advice indeed.

14 Acknowledgements

This presentation owes much to the ongoing research with John Plaice on the role of atomism in the shaping of modern science and society in general.

This work was supported in part by a grant from the Canadian Natural Sciences and Engineering Research Council (NSERC).