Knowledge Representation within Soft Systems Methodology: Some Observations on the Application of Nonmonotonic Logic to Soft System Methodologies Models

P.J. Houlihan[1]
R. Smith[2]
R.B. Watson[1]

[1]School of Information Technology
Swinburne University of Technology,
PO Box 218, Hawthorn, Australia 3122

[2]School of Management Information Systems
Deakin University,
Geelong, Australia 3217
Email: smithr@deakin.edu.au

Abstract

The work of Gregory, which renders the conceptual models developed in a Soft Systems Methodology (SSM) study in modal logic and so facilitates their use as a framework for knowledge elicitation, is described in outline. An example of the application of Gregory's approach to knowledge representation using nonmonotonic logic is reported. This example highlights three areas of concern. It is suggested that it might be possible to convert such representations into a framework based upon Poole's Default Logic, so addressing some of the concerns raised.

Keywords

Information Systems, Nonmonotonic Logic, Soft Systems Methodology

INTRODUCTION

The Soft Systems Methodology (SSM), developed by Checkland and colleagues at Lancaster University (UK) has been widely used for almost three decades, in the analysis of complex problem situations, including problems surrounding information systems provision. SSM is now the subject of a number of monographs (Checkland 1981; Wilson 1990; Checkland and Scholes 1990; Davies and Ledington 1991), and case studies and research papers addressing methodological matters have appeared.

One significant strand of research during the last decade has sought to clarify if (and how) SSM ideas and forms might better support the development of information systems (see for example Mingers (1992) and references therein). Notable amongst these efforts has been the work of Gregory (1992, 1993a, 1993b, 1993c, 1993d, 1995, with Merali, 1992, and with Lau, 1999) developing an approach called Logico-Linguistic Modeling. This work stands out because it specifies formally a means of proceeding from the less structured output of an SSM study to the rigorous structure demanded for the specification of computerised information systems. Other researchers (eg, Wilson 1989; Wood-Harper et al. 1985), have explored grafting or embedding SSM models into information systems analysis approaches which use a number of notations and concepts from traditional structured analysis (eg, data modelling), to interface subsequently to the specification of computerised information systems. Lewis (1993) has introduced the concept of "cognitive categories", derived from SSM conceptual models, as a means of establishing a front-end to information systems development. It could be argued, however, that none of these are as direct or rigorous as the set of concepts and the approach proposed by Gregory.

In the present paper we revisit Gregory's ideas, and highlight some possibilities for extending his work. Specifically, we provide a very brief introduction to SSM, establishing its relationship to Logico-Linguistic Models (LLMs). We also outline Gregory's approach for formalising models taken from SSM studies. This discussion, and an associated Appendix, demonstrates how a conceptual model can be developed to produce a LLM, that can be expressed in formal logic. Subsequently, we present an example of Gregory's approach to knowledge representation using nonmonotonic logic, and use this example to critique that work. We then suggest that some of the concerns which have surfaced in this examination might be addressed by converting LLM representations into a form based on Poole's Default Logic (PDL). Possible future research, explicitly the application of PDL in this context, is canvassed in the concluding sections.

THE LOGICO-LINGUISTIC APPROACH

Relationship to Soft Systems Methodology

Soft Systems Methodology has been described extensively in detail elsewhere (Checkland, 1981; Wilson, 1990; Checkland and Scholes, 1990; Davies and Ledington, 1991), so is not presented in any detail here. To situate the Logico-Linguistic approach however, it should be appreciated that SSM is a formalised version of purposeful thinking because it uses systems ideas to formulate statements about a perceived reality. In SSM, the statements about perceived reality take the form of "purposeful holons" (statements of purposeful human activities connected by arrows indicating that one activity is "logically contingent" upon another). These holons are referred to within the SSM literature as conceptual models. The recent work of Gregory renders the conceptual models developed in a SSM study in modal logic and so facilitates their use as a framework for knowledge elicitation and for the design of knowledge-based systems.

Outline of Logico-Linguistic Modeling

Gregory (1992) presents a six stage methodology for the creation of information systems using SSM as a basis. The six stages are:

1. *Systems Analysis* - this involves the conduct of a traditional SSM study, concluding with the development of various primary task conceptual models, relevant to a system which is to be implemented subsequently.

2. *Language Creation* - the output from this phase is a LLM. This model refines a selected conceptual model, produced at stage 1. It defines the system to be developed in a language, developed by stakeholders, that is expressible in first order predicate logic. The system defined by the LLM at this stage has the same relationship to the real world as the preceding conceptual model - i.e. it is purely notional.

3. *Knowledge Elicitation* - in this phase the LLM is enhanced by the inclusion of empirical or contingent definitions. This maps the LLM to the real world. It also leaves the enhanced LLM open to falsification as real world "facts" may change. At this stage the base language of the LLM is enhanced to include two modal operators. The modal operator **N** is used to indicate that the relation is a definition. The operator **C** is used to indicate that the relation is logically contingent upon empirical facts.

4. Knowledge Representation - the enhanced LLM is transformed into a modal predicate logic equivalent of the LLM.

5. Codification - at this stage the modal logic developed to represent the knowledge is transformed into a computer program. The language used by Gregory is Prolog.

6. Verification - this is an ongoing process, whereby new empirical information is incorporated into the model. This may cause the deletion of previously assumed contingent statements and the creation of new contingent deductions. It has been argued by Gregory (1995) that a system written in Prolog can be self verifiable and a proposal has been given by Gregory and Merali (1992) for an extension of Prolog that would allow the computer system to be self correcting.

An overview and example of the LLM approach, with emphasis on the pivotal stages 2 to 4, is reported in Appendix 1.

KNOWLEDGE REPRESENTATION USING NONMONOTONIC LOGIC

What is Nonmonotonic Reasoning?

Nonmonotonic reasoning is an attempt to formalise the commonsense reasoning that humans perform, often without conscious effort, in their everyday life. Some of the types of reasoning handled are:

- Where the information available is not complete or is contradictory and a rational decision has to be made;
- When implicit conventions are used when analysing or passing on information; and
- When a decision or inference made, based on incomplete information, has to be changed because more information becomes available.

Traditional logics are always *monotonic*. This can be roughly defined by the statement that, adding new premises to a set of premises does not invalidate old conclusions drawn from the original set of premises. Due to this property traditional 1ogics do not allow the representation of commonsense reasoning. Nonmonotonic logics come closer to common sense reasoning by allowing this rule to be violated. The rule is violated in a controlled way so that it is still possible to come to rational conclusions.

Gregory's Nonmonotonic Logic

In the second stage of Gregory’s methodology, a model is developed that is definitional in nature. That is, in the model derived from the LLM, any of the statements of the model are necessarily true. When the empirical rules from the real world are introduced in the third stage this is no longer the case, at least in the world view assumed by SSM. In this view it is proposed that it is not the real world that is systemic, rather it is the model of the real world that is systemic. Therefore, any facts that are introduced from the real world into the system are open to falsification. This might be because of an increase in knowledge about the real world, a change of circumstances in the real world or just because the real world turns out not to be systemic. Therefore, to represent this knowledge a language is needed that is open to falsification. Nonmonotonic logic provides this language.

In Gregory (1995), a nonmonotonic logic is proposed for the purposes of representing the knowledge captured in the enhanced LLM. The logic defines two new operators and four meta-rules for the generation of formulae.

The **N** operator is used to denote formulae that are 1ogicaly true; these are the formulae that were defined in stage two of the methodology as necessary conditions. The **C** operator is used to denote contingent truth. This operator will be used to represent the empirical formulae introduced in the enhanced LLM.

A schema is also defined for the derivation of formulae within this logic. The axioms and the rules of formulation and derivation of formulae are the same as for the predicate calculus and a modal system known as "S5" (Lukaszewicz 1990) with the additional meta-rules given below. The modal system "S5" provides a basic schema for modal logics where logically true conditions (**N**) are always preferred over those that are contingently true (**C**).

I. If X |- Y then N(X) |- N(Y)

II. If X |- Y then C(X) |- C(Y)

III. If X, Y |- Z then N(X), N(Y) |- N(Z)

IV. If X, Y |- Z then C(X), N(Y) |- C(Z)

An Example

Figure 1 is an example used by Gregory and Merali (1992) and shows a portion of an enhanced LLM.

Formulae 1, 2 and 3 are the axioms of the system derived from this enhanced LLM using the N and C operators:

1. N: $\forall x\ (A(x) \leftrightarrow (B1(x)\ \&\ B2(x))$
2. C: $\forall x\ (A(x) \leftrightarrow (C1(x)\ \&\ C2(x)\ \&\ C3(x)\ \&\ C4(x))$
3. C: $\forall x\ ((C1(x)\ \&\ C2(x)\ \&\ C3(x)\ \&\ C4(x)) \rightarrow A(x))$

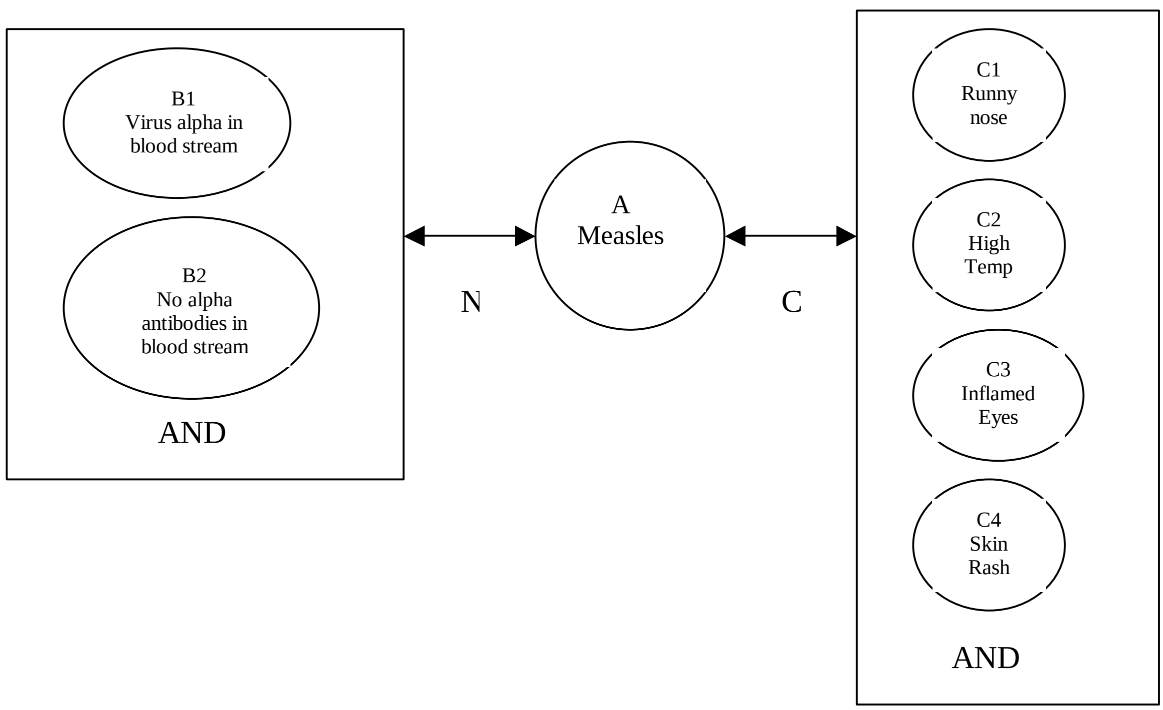

Figure 1: Measles diagnosis system LLM

From these, formulae 4 to 10 can be derived:

4. N : $\forall x\ (A(x) \rightarrow B1(x))$ [From 1]
5. N : $\forall x\ (A(x) \rightarrow B2(x))$ [From 1]
6. N : $\forall x\ ((B1(x)\ \&\ B2(x)) \rightarrow A(x)$ [From 1]
7. C : $\forall x\ ((C1(x)\ \&\ C2(x)\ \&\ C3(x)\ \&\ C4(x)) \rightarrow B1(x))$ [From 3 and 4]

8. C : $\forall x$ ((C1(x) & C2(x) & C3(x) & C4(x)) $\rightarrow$ B2(x)) [From 3 and 5]
9. C : $\forall x$ (A(x) $\leftrightarrow$ (C1(x) & C2(x) & C3(x) & C4(x)) [From 2 and 3]
10. C : $\forall x$ ((B1(x) & B2(x)) $\rightarrow$ (C1(x) & C2(x) & C3(x) & C4(x)) [From 2 and 6]

This schema is nonmonotonic because it allows the incorporation of new theorems that are incompatible with the existing ones. For example, if it were discovered that there is someone that has all four symptoms but does not have virus alpha in their bloodstream, this knowledge can be incorporated by adding formula 11:

11. N : $\exists$ x(C1(x) & C2(x) & C3(x) & C4(x) & $\neg$B1(x)).

Adding Formula 11 forces the withdrawal of Premise 3 and all formulae derived from this premise. To see this consider the following reasoning.

- Formula 11 is incompatible with Formula 7. The necessary formula must be preferred over the contingent formula therefore Formula 7 is false.
- As Formula 7 was derived from Formula 1 and Formula 3 the disjunction of these two statements is false.
- As the disjunction of Formula 1 and Formula 3 is false and because Formula 3 is contingent it must be false and therefore must be discarded.
- Therefore, all formulae derived using Formula 3 must be withdrawn; these are Formulae 8 and 9.

This leaves a system such that if a person is clinically diagnosed to have measles then it can be predicted that they will have the symptoms however, if they have the symptoms it is not possible to say whether or not they have measles.

Problems with Gregory's Nonmonotonic Logic

The example given above has highlighted a number of concerns:

1. As the above example shows, if there is one counter example to any contingent statement, then that statement must be removed completely from the system. As these statements have already been identified as contingent this approach may seem to be extreme.
2. The schema provides no way to decide between conflicting defaults. When a single person was found that did not conform to the axioms and premises it was possible to decide which one of Formulae 1 and 3 to exclude by favouring the necessary condition. If however, Axiom 1 is a contingent statement instead of a necessary one, this schema does not provide a way of deciding between the formulae. In the language of default logic (see next Section) there are two extensions: one where Formula 1 is assumed to apply and one where Formula 3 is assumed to apply.
3. When the counter example was found it became necessary to back track through the list of derived formulae, removing those that were no longer valid. Although in an automated system this might be possible, the analysis of the full consequences of removing derived formulae would be very complex and perhaps not possible.

A POSSIBLE RESOLUTION – POOLE'S DEFAULT REASONING

In addressing the concerns raised above, the present authors suggest that it might be useful to revisit the various categorisations of nonmonotonic reasoning. Some categorisations attempt to capture the difference in the underlying types of common sense reasoning that are being modelled (eg, Lukaszewicz 1990), while others base the classification on the differences in the formal 1ogics that have been developed (eg, Brewka 1991). In the present paper it is suggested that an approach due to Poole (1988), that would be classified by Lukaszewicz (1990) as default reasoning, may be useful.

Default reasoning is used when a rational conclusion is drawn from less than conclusive information and the conclusion that is drawn is at least plausible, because it does not contradict any other information. This reflects the real world where sometimes a decision has to be made based on partial knowledge. This type of reasoning is said to be *defensible*, meaning that any conclusion derived on the basis of incomplete information may be found to be wrong when more evidence is available.

To make a rational decision rules are often used that are known to be generally true but not necessarily always true: these rules are called *defaults*. The term default is used because these rules always apply unless more specific information is available. The inference pattern most often used for this type of reasoning is:

1. If A is true then B is typically true (this is the default rule).

2. I know A is true and I know nothing to lead me to believe that B is not true.

3. Therefore, I will assume B is true.

The typical example of default reasoning is the situation where "Tweety is a bird" is true and the question is asked "Does Tweety fly?". Without reference to other knowledge the answer would be "yes". The default reasoning rule used in this case is "Birds typically fly". If later it is found that Tweety is an emu and it is known that emus do not fly then the original conclusion must be withdrawn.

There are various formulations of default reasoning. Houlihan (1996) has suggested that a logical framework due to Poole (1988) might be particularly suited to addressing the concerns raised above with Gregory’s approach. Poole's Default Logic (PDL) is based on the contention that the inability of classical logic to handle nonmonotonic reasoning, rather than being a problem with the logic, is a problem with the way the logic is used. In developing the theory, Poole takes as his model a reasoning system where hypotheses are proposed, tested and possibly withdrawn. This model is then used to develop a theory of default reasoning that avoids the use of modal symbols (such as C and N) and extra inference rules. Poole contends:

"If one allows hypothetical reasoning then there is no need to define a new logic to handle nonmonotonic reasoning."

It is beyond the scope of the present paper to examine the formal representation of enhanced LLMs in PDL, instead of using Gregory’s nonmonotonic logic. It is noted however that some steps towards this have been explored and are to be the subject of a future paper by the present authors. In particular, building upon the unpublished work of Houlihan (1996) it is suggested that it is possible to convert the enhanced LLMs generated as part of Gregory’s methodology into a PDL framework through a relatively straightforward five step process (Identify and name predicates; Identify domains; Identify facts; Identify defaults; and Add real world facts). This effectively solves the problems raised above, and provides the immediate advantage of being able to draw on a well researched formalism and all the knowledge available from Poole’s work.

CONCLUSION

In this paper the work of Gregory, which renders the conceptual models developed in a SSM study in modal logic and so facilitates their use as a framework for knowledge elicitation and for the design of knowledge-based systems, has been described in outline. The major contribution of this paper has been the identification of some concerns with Gregory's approach to knowledge representation using nonmonotonic logic within an extended LLM. In particular this paper highlights issues surrounding:

- The extreme process by which, if there is one counter example to any contingent statement, then that statement must be removed completely from the system;
- The observation that the schema provides no way to decide between conflicting defaults; and
- The process of having to backtrack through the list of derived formulae, removing those that were no longer valid.

Further, the paper has suggested that it might be possible to convert LLM representations into a framework based upon Poole's Default Logic, and in so doing to address the concerns raised.

REFERENCES

Brewka, G. (1991) *Nonmonotonic Reasoning: Logical Foundations of Commonsense*, Cambridge University Press.

Checkland, P.B. (1981) *Systems Thinking Systems Practice*, Wiley.

Checkland, P.B. and Scholes, J. (1990) *Soft Systems Methodology in Action*, Wiley.

Davies, L. and Ledington, P. (1991*) Information in Action - Soft Systems Methodology*, Macmillan Information Series.

Gregory, F. (1992) SSM to Information Systems: A logical account, *Systemist* **14** (3) 180-189.

Gregory, F. (1993a) Soft systems methodology to information systems: a Wittgensteinian approach, *Journal of Information Systems* **3** (3) 149-168.

Gregory, F. (1993b) Cause, Effect, Efficiency and Soft System Models, *Journal of the Operations Research Society* **44** (4) 333-344.

Gregory, F. (1993c) Logic and Meaning in Conceptual Models: Implications for Information Systems Design, *Systemist* **15** (1) 28-43.

Gregory, F. (1993d) *Mapping Conceptual Models on to the Real World", in "Systems Science: Addressing Global Issues*, ed. Stowell, F. et al., Plenum, 117-122.

Gregory, F. (1995) Soft Systems Models for Knowledge Elicitation and Representation, *Journal of the Operational Research Society* **46** (5) 562-578.

Gregory, F.H., and Lau, S.P. (1999) Logical Soft Systems Modelling for Information Source Analysis – The Case of Hongkong Telecom, *Journal of the Operational Research Society* **50** 124-137.

Gregory, F. and Merali, Y. (1992) *Inductions, Modality and Conceptual Modelling*, Warwick Business School Research Paper, No. 79, March 1992.

Houlihan, P. (1996) *Knowledge Representation within SSM: The Application of Nonmonotonic Logic to Soft System Methodology models,* Master of Information Technology minor thesis, Swinburne University of Technology, January 1996.

Houlihan, P., Smith, R. and Watson, R.B. (1996) *Knowledge Representation within Soft Systems Methodology: An Examination of the Logico-Linguistic Approach*, 2nd Australian Systems Conference, Monash University, Melbourne Victoria.

Lukaszewicz, W. (1990) *Non-monotonic Reasoning: Formalization of Commonsense Reasoning*, Ellis Horwood series in Artificial Intelligence,

Lewis, P.J. (1993) Linking Soft Systems Methodology with Data-focused Information Systems Development, *Journal of Information Systems* **3** (3) 169-186.

Mingers, J. (1992) SSM and Information Systems: An overview, *Systemist* **14** (3) 82-88.

Poole, D. (1988) A Logical Framework for Default Reasoning, *Artificial Intelligence* **36** 27-47.

Wilson, B. (1989) A Systems Methodology for Information Requirements Analysis, in *Systems Development for Human Progress*, ed. H.K. Klein and K. Kumar, Elsevier Science (North Holland), 175-195.

Wilson, B. (1990*) Systems: Concepts, Methodologies and Applications*, Wiley.

Wood-Harper, T., Antill, L. and Avison, D. (1985*) Information Systems Definition: The Multiview Approach*, Blackwell Scientific Publications.

APPENDIX 1: LLM OVERVIEW

In Sections 2 and 3 of this paper we discuss some issues related to stages 2 to 4 of Gregory's LLM approach. Therefore, for completeness, a further discussion of these steps follows. Note that the example presented is adopted from a previous publication by the present authors (Houlihan et al. 1996).

Stage 2 - Language Creation

In this stage, one of the conceptual models that was produced as part of the SSM study is taken as a basis for further development. From this a LLM is produced. It is intended that this model be produced in the same way as a conceptual model, that is through an interactive debate that occurs between the stakeholders. LLMs are diagrams that look similar to conceptual models, however they are constructed in such a way as to be expressible in propositional or first order logic.

Figures A1 and A2 show a sample Conceptual Model (for a system to produce advertising campaigns) (Figure A1), transformed into a LLM (Figure A2). The transformation of a conceptual model such as this, into a LLM, is described by Gregory (1993a) in terms of four rules.

Rule 1: Convert commands into statements

Activities within conceptual models are expressed as imperative statements, that is, commands to perform an action. However, traditional logics, such as propositional and first order logic, cannot deal with imperative statements. Therefore the first rule is to convert all commands into statements. In the example, the activity "design campaign" becomes "campaign is designed".

Rule 2: Include conditions that are sufficient but not necessary

The arrows in conceptual models represent necessary conditions, that is conditions that must occur before the related activity can occur. In the advertising company example the condition "campaign is designed" is a necessary condition for "campaign is implemented". In causal terms "campaign is implemented" cannot happen unless "campaign is designed" has already happened. To these relations of necessity, relations of sufficiency are now added to allow the expression of options.

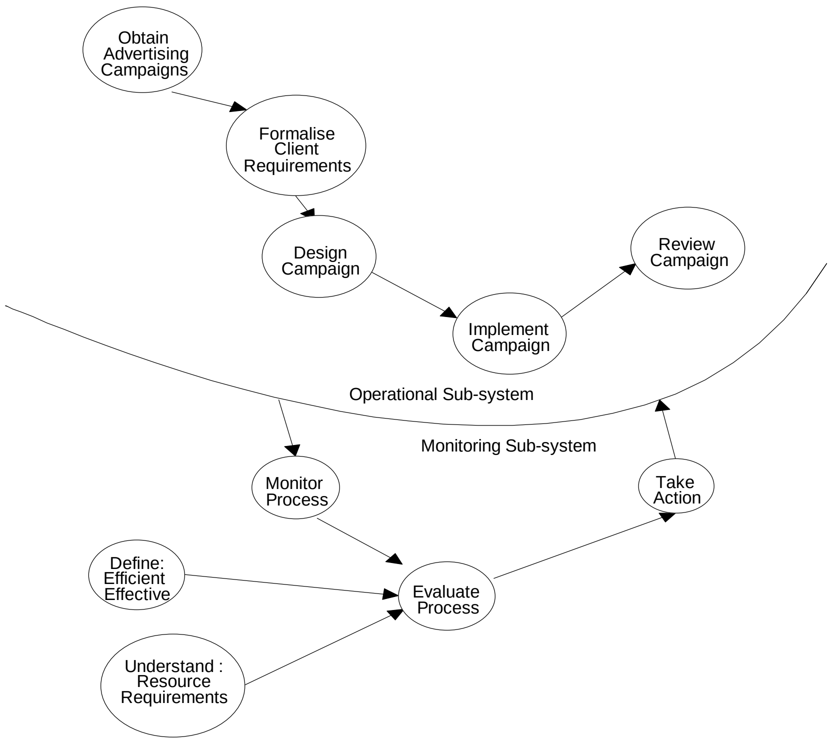

Figure A1: Conceptual Model: Advertising Company Example

The truth of a sufficient condition guarantees the truth of the dependent condition. However, the truth of a sufficient condition is not necessary for the truth of the dependent condition, there may be other conditions that imply the dependant condition. Hence they are referred to as sufficient but unnecessary (SUN) conditions.

Gregory uses several conventions for drawing SUN conditions on LLMs. In early papers the condition was shown by using a dotted line, however in later works they are drawn as individual conditions shown within a larger box with the word "OR" as a label in the box. We have adopted the latter convention. This represents that there may be many SUN conditions implying a dependent condition. In the example, the conditions "new campaign is designed" and "old campaign is reused" are SUN conditions with the dependent condition being "campaign is designed".

Rule 3: Ensure that all possible SUN conditions are included

Gregory distinguishes between two types of definitions: *extensive* and *intensive*. Extensive definitions specify all the members of a class and intensive definitions give the criteria for class inclusion. If the conditions of a LLM are taken to be classes, then the SUN conditions represent extensive definitions. Therefore, when SUN conditions are shown on the model it must be ensured that the list of conditions is complete, because these conditions will be taken to be a definition for the dependent condition.

In the example this means that "campaign is designed " is defined by the stakeholders to be the same as "new campaign is designed" or "old campaign is reused". Although it is possible to think of other ways of designing advertising campaigns, the system defined in this example is restricted to just these two ways.

Rule 4: Ensure that the set of necessary conditions is sufficient

In the example, the condition "campaign is designed" is a necessary condition for "campaign is implemented" but it is not a *sufficient condition*: it is possible for "campaign is designed" to be true without "campaign is implemented". The meaning of the conceptual model is certainly that if a campaign is designed then it is implemented. To force this idea of *causality* in the LLM, Gregory argues that new conditions must be included in the model.

In the example, the condition "an agent is employed who will implement a campaign that has been designed" is introduced. This condition, with the condition "campaign is designed" is a set of necessary and sufficient (N&S) conditions to establish the truth of "campaign is implemented". A set of N&S conditions is shown on a LLM in a similar fashion to a set of SUN conditions, except that the word "AND" is shown as a label for the box.

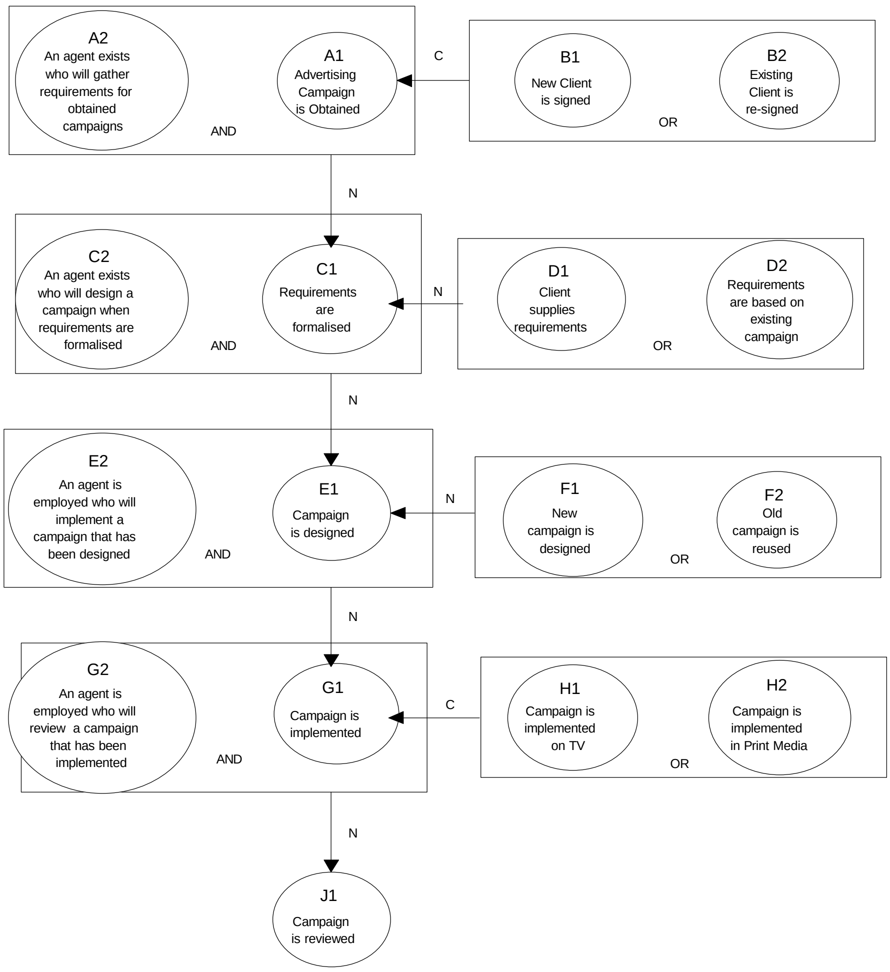

Figure A2: LLM: Advertising Company Example

Stage 3 - Knowledge Elicitation

The LLM generated in the previous stage bears the same relationship to the real world as conceptual models and all the system modelling that occurs during an SSM study. This means that there is no claim that the model represents the real world. Rather, it is an artificial systemic construct used to investigate the real world. However, to use these models to direct real world activity and to be able to construct computer systems, a way of establishing meaning for these models is required. This is done by linking the model to real world facts using modal operators.

The LLM generated consists of conditions that are necessarily true because that is how they have been defined. When linking these models to the real world it may be assumed that there will be some instances where the systemic model built so far does not match the real world. Statements are now introduced whose truth can be established in the real world outside the system model. Relationships between these statements and the statements already defined in the model are called contingent, because the truth of these relationships is contingent upon the real world. These conditions represent the link between the systemic model and the real world as defined by the stakeholders. They are shown on the LLM with arrows labelled with a **C**. The pre-existing arrows are now labelled with an **N**.

The use of the **N** and **C** identifiers is taken from the field of modal logic and the usual interpretation applied to modal operators is termed *possible world semantics*. In this interpretation statements may fall into two categories, those that are true in all possible worlds and those that need not be true in all possible worlds: necessarily true statements or contingent statements.

The use of these operators by Gregory is slightly different. Under the interpretation used for LLMs a statement that is necessarily true is so because it has been defined that way and no other interpretation is admitted. Whereas it is accepted that there may be an example that disproves a contingently true statement. Until a counter example is found, contingently true statements are assumed to be universally true. This definition of contingently true corresponds to the definition sometimes given for an *inductive hypothesis*.

In the example, the conditions "campaign is implemented on TV" and "campaign is implemented in print media" are introduced as contingent conditions, being real world definitions of "campaign is implemented".

Stage 4 - Representation in Formal Logic

The LLM can then be represented in propositional logic and by extension, in first order logic (A brief account of the symbols from first order logic used is present in Appendix 2). By ensuring that all the SUN conditions have been identified and that the set of necessary conditions is sufficient, the LLM represents equivalence relations in propositional logic. The example shown above generates the following set of statements in propositional logic:

1. $C : (B1 \vee B2) \quad \leftrightarrow \quad A1$
2. $N : (A1 \,\&\, A2) \quad \leftrightarrow \quad C1$
3. $N : (D1 \vee D2) \quad \leftrightarrow \quad C1$
4. $N : (C1 \,\&\, C2) \quad \leftrightarrow \quad E1$
5. $N : (F1 \vee F2) \quad \leftrightarrow \quad E1$
6. $N : (E1 \,\&\, E2) \quad \leftrightarrow \quad G1$
7. $C : (H1 \vee H2) \quad \leftrightarrow \quad G1$
8. $N : (G1 \,\&\, G2) \quad \leftrightarrow \quad J1$

APPENDIX 2: LOGIC DEFINITION AND TERMS

This section provides a very brief account of the symbols from first order logic that are used in this paper and in associated papers referenced herein. For more detail, Lukaszewicz (1990) provides a good introduction to classical first order logic and its relationship to nonmonotonic logic.

The Alphabet

The language used consists of primitive symbols from the following pairwise disjoint classes:

i A denumerable set of *individual variables*. Shown as x, y, z, etc.

ii One truth constant: *True*.

iii Sentential connectives: $\neg$ and $\rightarrow$.
iv One quantifier: $\forall$.
v Various punctuation marks for clarity.
vi A countable set of *predicate constants*. To each predicate constant there is a uniquely assigned non-negative integer called its *arity*.
vii A set of object constants.

The sentential connectives &, $\vee$, $\leftrightarrow$ are used as short hand notation with the normal meaning. The existential quantifier, $\exists$, is used as it is normally defined. Any well formed formula from this language is simply called a formula.

Some Definitions

The *terms* of the language are all the individual variables and object constants. The set of atomic formulae consists of all the terms of the language together with the set:

> $\{P(x_1,...x_n)$: P is an n-ary predicate constant and $x_1,...x_n$ are terms of the language$\}$.

For any formula of the form $\forall xP(x)$ where x is a variable, x is said to be *universally quantified*.
For every occurrence of x in a formula P, x is said to be *universally bound* in $\forall xP$.
The definitions of *existentially quantified* and *existentially bound* are formed in a similar way. When there is no distinction between the type of quantifier then x is just said to be bound or quantified.
The occurrence of a variable, x, in a formula, P, is said to be *free* if it is neither bound nor quantified.
A formula containing no free variables is said to be *closed*, otherwise it is said to be *open*.
For an open formula, P(x), the *ground* instances of this formula with respect to a set of constants, $\{a_1,...a_n\}$, is the set of closed formulae $\{P(a_1),...,P(a_n)\}$.
For a set of formulae, T, then it is written that T |- P, if P is provable from T using the normal deduction system for classical first order logic.
The notation, Th(T), is used to stand for the set {P: T |- P }.

COPYRIGHT