AMCIS 1995 Proceedings

Americas Conference on Information Systems (AMCIS)

8-25-1995

A LOGIC THEORY BASED APPROACH TO INFORMATION SYSTEMS DECOMPOSITION

Rakesh R. Neunaha
Southern Illinois University at Carbondale, rakesh2@siu.edu

Recommended Citation

Neunaha, Rakesh R., "A LOGIC THEORY BASED APPROACH TO INFORMATION SYSTEMS DECOMPOSITION" (1995). *AMCIS 1995 Proceedings*. 6.
http://aisel.aisnet.org/amcis1995/6

A LOGIC THEORY BASED APPROACH TO INFORMATION SYSTEMS DECOMPOSITION

Rakesh R Neunaha, Department of Management,
Southern Illinois University at Carbondale, IL 62901-4627.
Ph: (618) 453-7887 E-Mail: rakesh2@siu.edu

Introduction:

Decomposition is a central concept to most systems analysis and design methodologies. Structured Analysis [DeMarco 1979], Warnier-Orr Diagrams [Orr 1977; Warnier 1974], Jackson System Development [Jackson 1983], and Higher Order Software [Hamilton and Zeldin 1976] are all based on the construction of a hierarchical structure for a proposed computer-based system.

Decomposition is *the breakdown of a complex system into smaller, relatively independent units, so as to simplify the construction of complex man-made systems*. Yourdon [Yourdon and Constantine 1979] observes that modular programs are 'easier to write, debug, maintain, and manage'. Decompositions that exhibit high cohesion within modules and low coupling between modules are considered appropriate. Decomposition methodologies can be used either to analyze an existing system [Simon and Ando 1961; Courtois 1985] or as a basis for the design of a new system [Simon 1981].

Though system decomposition is an integral part of the systems analysis and design process, it relies greatly on an analyst's experience and expertise. The lack of formal approaches for generating good decompositions greatly inhibits the systems analysis and design process. In this paper we present a logic theory-based approach to IS decomposition. The method is based on the ontological model of IS decomposition proposed by Wand and Weber [1988a; 1988b; 1989a; 1989b; 1990a; 1990b]. It incorporates elements of combinational logic theory to provide a simplified approach to generating 'good' decompositions. In cases of incomplete specifications, this approach also provides indicators to the areas and elements where more specific information is required.

The Ontological Approach to System Modelling:

Wand and Weber [1988a; 1988b; 1989a; 1989b; 1990a; 1990b] suggested a model of systems decomposition, based on Bunge's ontological model [1977; 1979]. A detailed formalization of this model is presented in Wand and Weber [1990a]. In order to provide a context for this work, we present key tenets of the Wand and Weber decomposition model:

Premise 1: The world is made of *objects* or *things* - *simple* and *composite* - that possess *properties* . Some properties of an object are of interest or observed. This set of properties yields a *functional schema* $F = \{ F1, ..., Fn \}$, where each function Fi is the value of an observed property i at time t.

The function Fi is essentially a state variable of the object, and the value of the function $F = \{ F1, ..., Fn \}$at a given time is the state of the object. The set $S(X)$of all such states is the possible state space of the object, and is equivalent to

$$S(X) = \{ (x1, ..., xn) \mid xi = Fi(t) \}$$

The dynamics of an object are defined in terms of its change of states:

An ordered pair of states $e = s\ s^*$, where $s, s^*\ S(X)$ is termed an *event*. The set of possible states and possible events are defined as lawful states and lawful events respectively. An event is either *external* (interaction with the environment - a disturbance) or *internal* (intrinsic - a reaction to disturbance) to the system.

Premise 2: There exist two kinds of states - stable and unstable states. A *stable state* is one from which a transition can occur only due to an external event either to a stable or to an unstable state. An *unstable state* is one, a transition from which must occur, either due to an external or an internal event, and finally to a stable state.

Premise 3 (Stability): An object in an unstable state will eventually attain a stable state.

An object is said to be well-behaved ***iff*** for every unstable state, the final stable state reached by the object is unique.

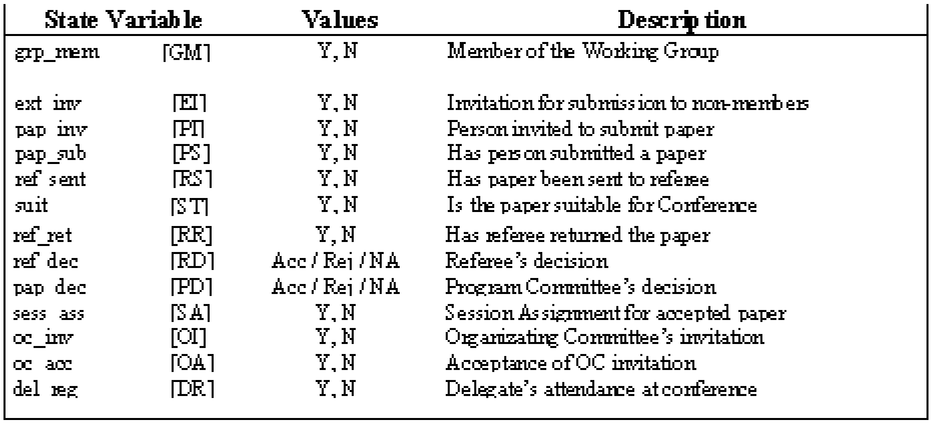
A *system* is a composite or an aggregate of objects, with *interactions* between all possible sub-systems. The set of objects in a composite object constitutes its *composition* . The composition and interaction set of a sub-system is a subset of the composition and interaction set of the super-system respectively.

A *decomposition of a system* is then the set of sub-systems $D() = \{ i \}$, such that each composition element in the system is included in atleast one sub-system, and each interaction is included as an interaction within / between the several sub-systems. There are three types of decomposition: sequential, conditional, and parallel. A *good decomposition* is one in which all sub-systems are well-behaved under their internal events that are induced by the corresponding external event set. There are several heuristics that serve the decomposition process [Paulson and Wand 1992].

A system specification is *the set of all state variables, external events, and sublaws (identified through the process of system analysis) which describe a real system.* The system must conform to a set of *stability conditions*. If the system conflicts with a stability condition after any event, it must take *corrective actions* and return to a stable state. The system specification is *incomplete* if a corrective action does not exist for an unstable state, and the is *inconsistent* if the corrective action leads to more than one stable state.

The Logic Theory Approach to System Decomposition:

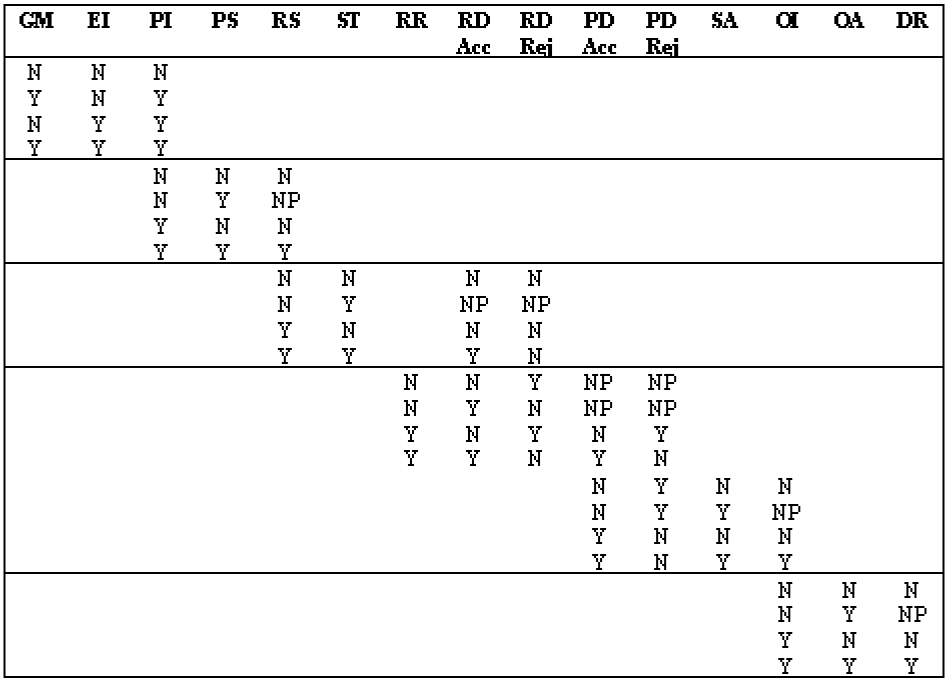
Based on this ontological model and formalized approach to decomposition, Paulson and Wand [1992] proposed the States, Events, and Laws Modeling Approach (SELMA) for analyzing a real system's specifications and to generate IS structures. We present a combinational logic theory-based approach and illustrate it using Paulson and Wand's example of the IFIP Working Group Conference problem - a situation analogous to two organizational entities within a corporate structure using common information.



State Variable		Values	Description
grp_mem	[GM]	Y, N	Member of the Working Group
ext_inv	[EI]	Y, N	Invitation for submission to non-members
pap_inv	[PI]	Y, N	Person invited to submit paper
pap_sub	[PS]	Y, N	Has person submitted a paper
ref_sent	[RS]	Y, N	Has paper been sent to referee
suit	[ST]	Y, N	Is the paper suitable for Conference
ref_ret	[RR]	Y, N	Has referee returned the paper
ref_dec	[RD]	Acc / Rej / NA	Referee's decision
pap_dec	[PD]	Acc / Rej / NA	Program Committee's decision
sess_ass	[SA]	Y, N	Session Assignment for accepted paper
oc_inv	[OI]	Y, N	Organizating Committee's invitation
oc_acc	[OA]	Y, N	Acceptance of OC invitation
del_reg	[DR]	Y, N	Delegate's attendance at conference

Table 1: Definition of State Variables for the IFIP Conference Problem

Each bi-valued state variable can be treated as a logic variable, while an n-valued (where n > 2) state variable (eg., RD, PD) could be converted into (n-1) dummy logic variables (eg., RD=Acc and RD=Rej). We could then generate a combinational logic table (or truth table) for this set of logic variables to present all possible (lawful) states and events. Some of these (state) variables could then be treated as intermediate and/or final dependent variables. We could then derive logic functions for them to yield the corresponding decomposed functions. This is similar to the approach taken in the design of logic networks and circuits.



GM	EI	PI	PS	RS	ST	RR	RD Acc	RD Rej	PD Acc	PD Rej	SA	OI	OA	DR
N	N	N												
Y	N	Y												
N	Y	Y												
Y	Y	Y												
		N	N	N										
		N	Y	NP										
		Y	N	N										
		Y	Y	Y										
				N	N		N	N						
				N	Y		NP	NP						
				Y	N		N	N						
				Y	Y		Y	N						
						N	N	Y	NP	NP				
						N	Y	N	NP	NP				
						Y	N	Y	N	Y				
						Y	Y	N	Y	N				
									N	Y	N	N		
									N	Y	Y	NP		
									Y	N	N	N		
									Y	N	Y	Y		
												N	N	N
												N	Y	NP
												Y	N	N
												Y	Y	Y

Table 2: Combinational Logic Table for the IFIP Conference Problem

In the above combinational logic table, we have separated the set of variables into smaller sets in order to reduce the complexity of the table. All states are specified. Whenever a particular state case is not possible (illegal), it is indicated as NP (not possible) under the intermediate logic function column. One can now construct logic functions for each one of the intermediate functional states:

PI = GM + EI PI : F(GM; EI)

RS = PI * PS RS : F(PI; PS)

RD(Acc) = RS * ST RD : F(RS; ST)

PD(Acc) = RD(Acc) * RR PD : F(RD; RR)

OI = PD(Acc) * SA OI : F(PD; SA)

DR = OI * OA DR : F(OI; OA)

Logic Functions for the Intermediate and Final Dependent State Variables

This decomposition is a concise one and is much easier to arrive at than with the approach suggested by Wand and Weber. The use of logic theory also enables us to easily focus on elements with incomplete information, since the decision variable in that area will be computationally incomplete. The logic theory approach is most suitable when used with binary-value state variables. But this can be easily extended to incorporate multi-valued state variables through the use of dummy state variables, one for each value the variable takes. A rudimentary illustration of this is provided in the case of the RD and the PD state variables in this example. This approach needs to be further refined in order to enhance its application to system decomposition problems.

(References available upon request)