DISCUSSION PAPER NO. 350

A STRUCTURE FOR COMPUTER-AIDED CORPORATE PLANNING

by

J.-M. Blin,[1] Edward A. Stohr[1]
and Mohan Tanniru[2]

Northwestern University
Graduate School of Management
Evanston, Illinois 60201

September, 1978

To appear in Policy Analysis and Information Systems, Vol. 2 , No. 2, December, 1978.

[1]Northwestern University, Graduate School of Management

[2]University of Wisconsin, Madison, School of Business Administration

ABSTRACT

This paper describes the design of a decision support system which provides a framework in which planning models covering different aspects of a corporation's activities can be integrated with a central financial plan. A linear hierarchical data model is used to describe the firm's chart of accounts. Once this 'schema' has been defined, the interactions of the various planning models with the firm's chart of accounts are described by 'subschemas' associated with each model. The paper develops a 'structured' approach to defining planning models and derives a 'standard representation' for an important class of models. It is shown that the 'standard representation' can be used to generate the data for a number of different operations research algorithms. The techniques developed in the paper are illustrated by describing their application to a production planning model and a long-range financial planning model.

Introduction

Economists, management scientists and computer scientists have produced many different planning models. However, little attention has been given to the integration of these models into a comprehensive, easily accessible and computer implementable overall model of a firm. There are many computer packages which implement a basic accounting structure to enable the manager to generate summary reports of his company at various target dates. But these models rarely attempt to integrate optimization methods or other analytical tools to evaluate company performance under alternative corporate policies. A successful integration of these features in a corporate planning model would enable the manager to make full use of the speed of execution, accuracy and convenience afforded by computers in other planning applications. This paper proposes a powerful self-contained framework to achieve this goal. Its usefulness is demonstrated by analyzing some representative corporate planning models.

The organizational setting we assume is as follows. The planning system is designed to support management decision-making and to help in the coordination of the budgets for various organizational units. Model builders and system analysts are responsible for the logical design and maintenance of the system. The planning system contains a central module called the 'central financial model' (CFM) which at any time contains the financial and budgeting information for the firm in a manner consistent with the firms chart of accounts. The CFM can be used to display the past history of various financial and economic time series. It also contains information concerning the adopted budgets and plans and has the capability of automatically projecting proforma financial statements by time-series and/or regression methods. Various simulation and operations research models can be used in conjunction with the forecasting models to arrive at a set of 'official' projections.

The planning system also contains modules which support the planning function at a more detailed level. These are categorized by functional area and/or organizational units. For ease of reference these modules will be called 'functional area models' (FAMS). It should be pointed out that we use the word 'model' in its broadest sense to mean any subsystem which provides computerized inputs to the CFM. This might include, for instance, subjectively estimated plans and budgets produced by lower organizational units.

The planning system is designed to assist the manager by providing: 1) retrieval of historic and projected information concerning the state of the organization, 2) integration of the plans of the various functional units in the organization allowing both 'top-down' and 'bottom-up' planning, and 3) automatic projection of proforma financial statements. It assists the model builder (system analyst) by providing: 1) a simple method of expressing the logic of the planning system as data which can then be operated on to provide different configurations of multiple models with the output of one model being used as input to other models, 2) a comprehensive system for maintaining the interconnections between models and allowing their inputs and outputs to be checked for consistency with the overall financial plan of the organization, 3) information concerning all data elements and models in the planning system.

Given the broad scope of the system to be described and to make the paper self-contained, portions of this paper summarize some key ideas which have appeared in [2], [11], [15]. This paper is organized in five sections. Following this introduction, section one outlines the structure of the decision support system (DSS) currently being implemented. Section two describes the linear hierarchical data model representing the accounting system. Section three demonstrates the integration of FAMS and CFM. Section four develops a standard representation of various models that the corporate planner may wish to use. Finally, section five illustrates the translation of a simple corporate planning model [10], into this standard system.

I. The Decision Support System (DSS) [1]

The planning system is currently being implemented in APL. Its major software components are:

1. The System Manager (SM)

This contains the software for generating, managing, storing and retrieving the accounting-related logic of the corporate planning models (both CFM and FAMS). This module is based on the linear hierarchical model which is outlined in Section II. This subsystem is the main vehicle for transferring data between the CFM and FAMS.

2. Model Statement Generation Subsystem (MSG).

This module is designed to help the model-builder by automating the logical statement of planning models and eliminating the necessity to include accounting identities in the user specification of the model. A 'standard representation' of planning models is developed and stored by this module as discussed in Section IV.

---

[1] See Alter [1] and Gorry and Scott-Morton [7] for discussions of decision support system requirements.

3. The Model Management Subsystem (MM)

This module provides interfaces between various algorithms (such as linear programming, goal programming and linear-quadratic control theory algorithms), and the 'standard representation' of the model stored by the MSG subsystem. It also utilizes database techniques to assist in running the model and managing the inputs and results during sensitivity analyses (see Tanniru [15]).[2]

4. Management Science Algorithms and Simulation Planning Language

As mentioned above, various algorithms can be used to implement the users model under the control of the MM subsystem. The simulation planning language (see Stohr [13]), is called by the MSG subsystem to help define the model as demonstrated in Section V below. It can also be used to solve the model if simulation is the preferred technique. Its capabilities are similar to those of a number of similar languages [4], [6].

5. Forecasting Module (Time-series and Regression)

6. Display and Support Facilities

   (a) Data Base Management System (DBMS)

   (b) Planning System Data Dictionary

   (c) Report Generator

   (d) Graphics Package

   (e) Utility Routines

In this paper we will concentrate on a description of the MSG subsystem. The SM subsystem was described in [11] and the MM subsystem will be described in a subsequent paper.

II. The Linear Hierarchical Data Model

Accounting transactions are used to report the monetary effects of different activities within the firm and in the final analysis financial statements provide the yardstick for measuring its performance.

---

[2] See Bonczek, Holsapple and Whinston [3] and Haseman [8] for other applications of data base management techniques to planning systems.

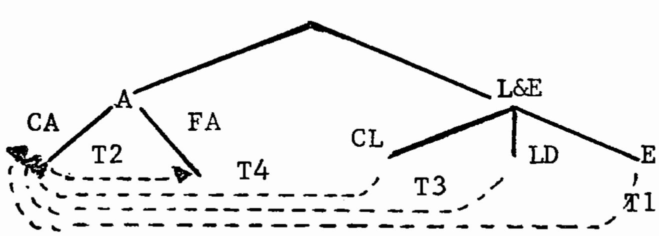
Tree diagrams provide a well-known means for representing hierarchical systems. Clearly, double entry bookkeeping can be formalized by such a tree. Given some initial resources the basic distinction is between their sources on the one hand--owned or borrowed--and their uses on the other hand. The former are customarily known as the equity and liability of the corporation; while the latter constitute its assets. Together they form the firm's balance sheet which describes the state of the system. Changes in the system affect the levels of the balance sheet accounts. These are continuously updated to reflect events in the corporation. For performance evaluation, it is also useful to classify the events by their effects (inflows or outflows) on the state of the system. A standard classification is given in the 'Income Statement'.

These considerations can be summarized in a tree graph. The nodes describe elements in the classification scheme and the flows are represented by directed arcs connecting leaf nodes. Figure 1 represents a basic breakdown of a corporate balance sheet. Obviously to affect a node requires a transaction connecting <u>two</u> leaf nodes in the subtree associated with that node. This means that a viable algebraic representation of this tree requires (1) a <u>matrix</u> with a positive and an offsetting negative entry to account for transactions affecting leaf nodes and (2) a <u>vector</u> with as many entries as there are conceivable links between leaf nodes. What is conceivable depends on the classification chosen and the legal bounds imposed upon the corporation. Certain transactions may not be feasible either from a legal or accounting viewpoint.

<u>The 'Systems Matrix'</u>

The effect on the leaf nodes can be summarized by the conventional node-arc 'incidence matrix'. If one also wants to represent the effect on higher order nodes in the tree structure, it suffices to add a row for each such node. Each transaction linking two leaf nodes defines a unique loop in the tree and will correspond to a column of the matrix. Thus a complete algebraic representation of the tree requires a (0,+1,-1) matrix with as many rows as there are nodes in the tree and as many columns as there are feasible transactions between leaf nodes. Finally we need to adopt a sign convention for the direction of the transaction. Our convention is to take debit entries--i.e. increases in the assets accounts or decreases in the equity/liability accounts--as positive and credit entries--i.e. decreases in the assets accounts or increases in the equity/liability accounts--as negative. The resulting matrix will henceforth be referred to as the (m X n) 'systems matrix' S. Corresponding to Figure 1:

$$S = \begin{array}{c} \\ A \\ L\&E \\ CA \\ FA \\ CL \\ LD \\ E \end{array} \begin{array}{c} \begin{array}{cccc} T_1 & T_2 & T_3 & T_4 \end{array} \\ \begin{bmatrix} +1 & 0 & +1 & +1 \\ -1 & 0 & -1 & -1 \\ +1 & -1 & +1 & +1 \\ 0 & +1 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix} \end{array}$$



Legend

Balance Sheet	Transactions
Current Assets (CA)	T1 - investment in firm by owners
Fixed Assets (FA)	T2 - purchase of fixed assets for cash
Current Liab. (CL)	T3 - long-term borrowing
Long-term Debt (L)	T4 - short-term borrowing
Equity (E)	

Figure 1

ILLUSTRATIVE DATA MODEL

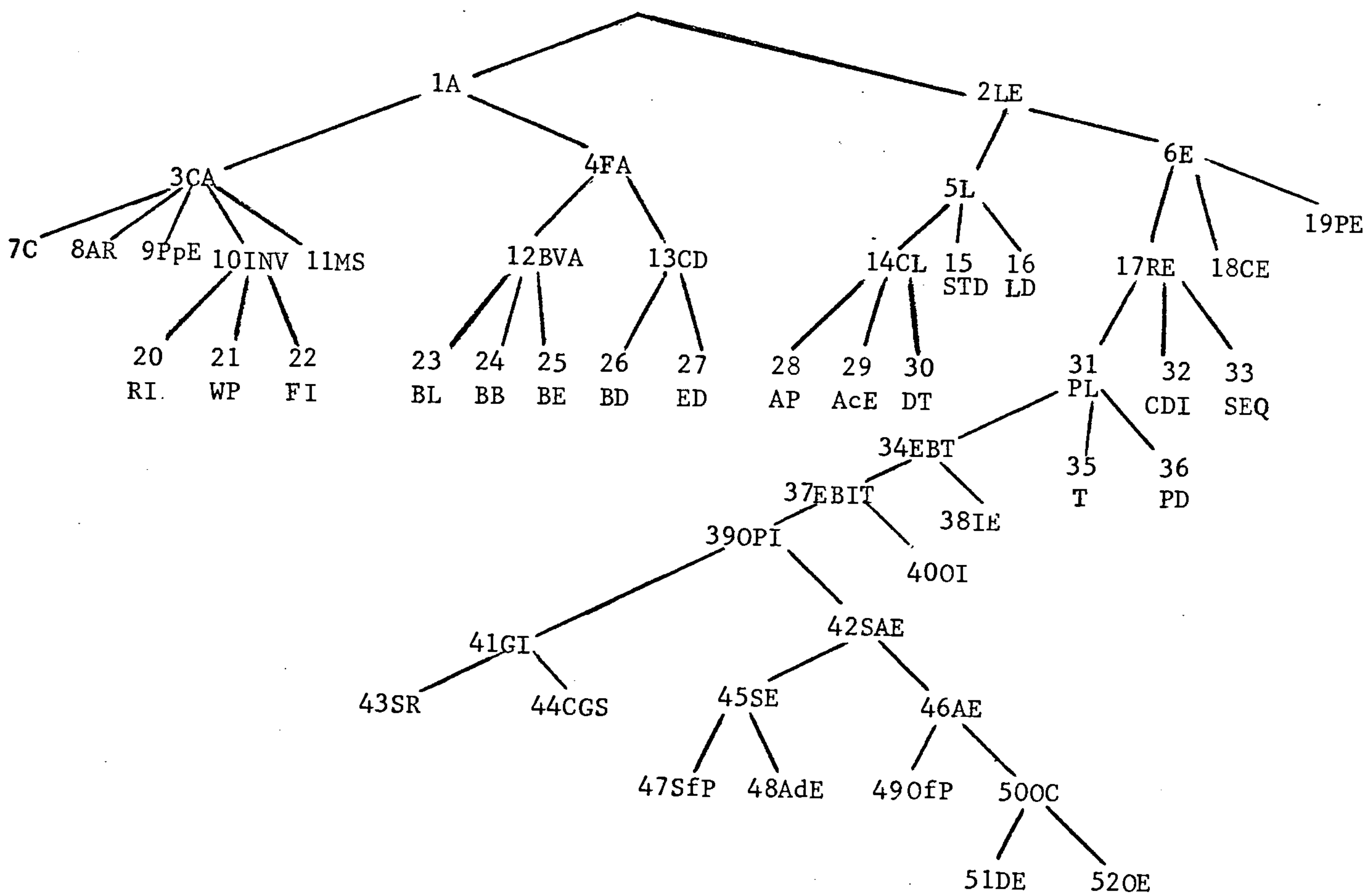
The 'Aggregate Transactions Vector'

As noted earlier, the states of the nodes of the system are changed by transactions reflecting corporate decisions and activities. States are observed at discrete points--perhaps mandated by law. Between two observations (t,t+1) the amount of each type of transaction is recorded as a separate entry in the aggregate transactions vector, τ. For instance, in our previous example we could have $\tau^t \in \mathbb{R}^4$ where τ_1^t denotes the aggregate investment by the firm's owner(s) during the t^{th} period (transaction type T_1) etc.

The matrix multiplication $S \cdot \tau$ summarizes the change in the state of the system. Denoting by $b^t \in \mathbb{R}^m$ the state of the system (values associated with nodes in the tree), the transition equation for the system is:

$$b^{t+1} = b^t + S \cdot \tau^t \qquad (1)$$

Figure 2 represents a more detailed standard financial schema.



TREE REPRESENTATION OF A CORPORATE FINANCIAL REPORTING STRUCTURE

Figure 2

Figure 2 (continued)

Account (node)	Symbol	Description
1.	A	Assets
2.	LE	Liabilities & Equities
3.	CA	Current Assets
4.	FA	Fixed Assets
5.	L	Liabilities
6.	E	Equities
7.	C	Cash
8.	AR	Accounts Receivable
9.	PpE	Pre-Paid Expenses
10.	INV	Inventory
11.	MS	Marketable Securities
12.	BVA	Book Value of Assets
13.	CD	Cumulative Depreciation
14.	CL	Current Liabilities
15.	STD	Short-term Debt
16.	LD	Long-term Debt
17.	RE	Retained Earnings
18.	CD	Common Equity
19.	PE	Preferred Equity
20.	RI	Raw Materials Inventory
21.	WP	Workin-in-Process
22.	FI	Finished Inventory
23.	BL	Book Value of Land
24.	BB	Book Value of Buildings
25.	BE	Book Value of Equipment
26.	BD	Depreciation on Land & Bldg.
27.	ED	Equipment Depreciation
28.	AP	Accounts Payable
29.	AcE	Accrued Expense
30.	DT	Deferred Taxes
31.	PL	Profit and Loss
32.	CDI	Common Dividend
33.	SEQ	Equity Sale
34.	EBT	Earnings Before Taxes
35.	T	Taxes
36.	PD	Preferred Dividend
37.	EBIT	Earnings Before Interest & Taxes
38.	IE	Interest Expense
39.	OPI	Operating Income
40.	OI	Other Income
41	GI	Gross Income
42.	SAE	Sales & Administrative Expenses
43.	SR	Sales Revenue
44.	CGS	Cost of Goods Sold
45.	SE	Sales Expense
46.	AE	Administrative Expenses
47.	SfP	Sales Force Expense
48.	AdE	Advertising Expense
49.	OfP	Office Payroll
50.	OC	Other Costs
51.	DE	Depreciation Expense
52.	OE	Other Expenses

A TYPICAL LIST OF ACCOUNTS

Figure 2 (concluded)

NO.	Description of Transactions	Node No. (from-to)	Acct. Entries (Cr./Db.)
1.	Cash Sales	43,7	SR/C
2.	Sales on Account	43,8	SR/AR
3.	Sales Force Expense	7,47	C/SfP
4.	Advertising Expense	7,48	C/AdE
5.	Cost of Goods Sold	22,44	FI/CGS
6.	Collections of Receivables	8,7	AR/C
7.	Office Salary Expense	7,49	C/OfP
8.	Labor Costs Paid in Cash	7,21	C/WP
9.	Material Used in this Period	20,21	RI/WP
10.	Direct Overhead	7,21	C/WP
11.	Pre-paid Expenses Adjusted For	9,52	PpE/OE
12.	Depreciation Expense	26,51	BD/DE
13.	Inventory Increase	21,22	WP/FI
14.	Cash Purchase of Inventory	7,20	C/RI
15.	Purchase of Inventory on Account	28,20	AP/RI
16.	Payments of Accounts Payable	7,28	C/AP
17.	Equipment Bought for Cash	7,25	C/BE
18.	Purchase of Equipment on Account	28,25	AP/BE
19.	Indirect Expenses	7,52	C/OE
20.	Repayment of Accrued Expense	7,29	C/AcE
21.	Payment of Deferred Taxes	7,30	C/DT
22.	Revenue from Marketable Sec.	40.7	OI/C
23.	Interest Expense	7,38	C/IE
24.	Taxes Paid in Cash	7,35	C/T
25.	Taxes Deferred	30,35	DT/T
26.	Other Expenses Deferred	29,52	AcE/OE
27.	Preferred Dividend Paid in Cash	7,36	C/PD
28.	Common Dividend Paid in Cash	7,32	C/CDI
29.	Expenses Pre-paid (Rent, etc.)	7,9	C/PpE
30.	Repayment of Long-term Debt	7,16	C/LD
31.	Repayment of Short-term Debt	7,15	C/STD
32.	Proceeds from Short-term Loan	15,7	STD/C
33.	Purchase of Marketable Securities	7,11	C/MS
34.	Proceeds from Sale of Marketable Sec.	11,7	MS/C
35.	Proceeds from Issue of Common Stock	18,7	CE/C
36.	Proceeds from C.E. in Excess of Par	18,33	CE/SEQ
37.	Proceeds from Issue of Preferred Stk.	19,7	PE/C
38.	Proceeds from Issue of L.T. Debt	16,7	LD/C

A TYPICAL LIST OF TRANSACTION TYPES

At this point it should be noted that we have expanded the tree in Figure 2 to include the income statement classification scheme. As we can recall this describes the events which lead to changes in the stock or balance sheet variables. For instance a cash sale of goods is associated with two transactions; namely, #1 and #5. The system consisting of the nodes corresponding to stock variables and nodes corresponding to a classification scheme for events can now be represented by a linear system as follows:

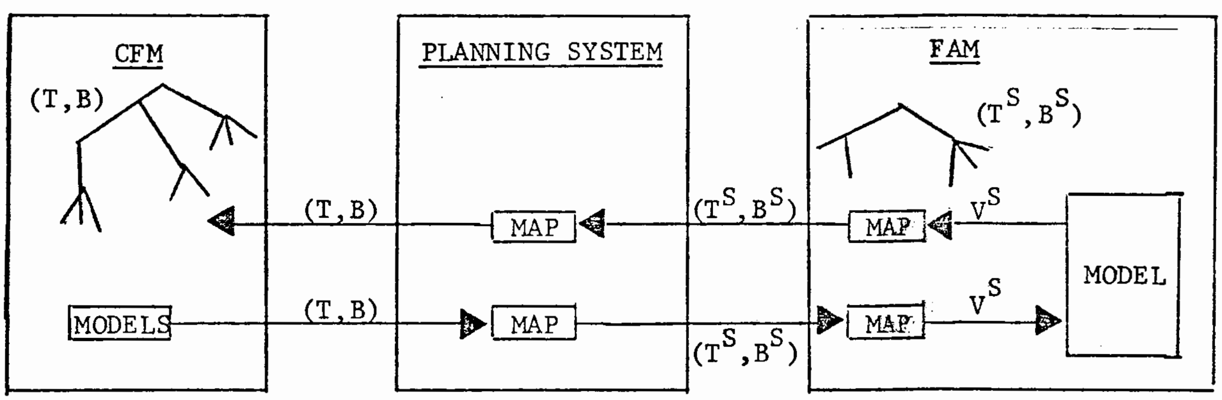
$$b^{t+1} = E \cdot b^{t} + S \cdot \tau^{t} \tag{2}$$

Here the (expanded) state space corresponds to all nodes in the tree. The (m X m) E matrix is an (m X m) identity matrix except that diagonal elements corresponding to non-balance sheet accounts are set equal to zero in order to respect the flow nature of the income statement accounts.

This completes the description of the graphical and algebraic representation of the accounting system employed by the SM subsystem. Borrowing terms from the database management field, the tree structure and list of arcs is a 'data model' and the firm's chart of accounts expressed in this way a 'schema'. Models introduced by planners to solve particular problems will be concerned with a subset of nodes and transactions. In fact the viewpoint of a particular model may require a rearrangement of nodes and a redefinition (aggregation or disaggregation) of transactions. The system's tree and set of arcs used by a model corresponds to the model's 'subschema' in database terms. An essential step in the procedure used to reconcile a model's outputs with the financial accounting system is to provide a mapping between the model's subschema and the official schema. A partly automated procedure for doing this is outlined in [11]. The system trees are stored internally using a 'multi-attribute' tree structure [14]. The systems matrices corresponding to the schema and subschemas are not stored but can be generated as required for use by the models. One immediate implication of (2) is that it contains all relevant accounting identities. If the initial state of the system is known and the aggregate transactions vector τ_t can be estimated, then the projected financial statements can be immediately computed using (2). As another application, in simulation models the modeler usually has to write a large number of statements describing accounting identities. Since these are already stored in the schema this task can be eliminated as described in Section IV.

III. Integrating Functional Area Models (FAMS) with the Central Financial Model (CFM)

A common distinction in the planning literature is between the 'top-down' and 'bottom-up' approach to planning. The top-down approach 'maps' the CFM into FAM's by disaggregating high-level plans and budgets into increasingly detailed plans--both functionally and temporally. Conversely the bottom-up approach aggregates detailed plans and budgets into global plans at the CFM level. This 'unfolding' or 'folding' develops targets for future states of the system. Since the targets must be achieved by specific transaction levels (τ) we use transactions rather than states as the primary link between models at different levels. Figure 3 represents the logic of the top-down and bottom-up mappings.



Legend: T (T^S) = set of transaction types--arcs, for the corporate schema (subschema).
B (B^S) = set of state variables--nodes in the tree-for the corporate schema (subschema).

Figure 3 - RELATIONSHIP BETWEEN CFM AND A FAM

A typical CFM comprises the following software and data:

1. Logical statements of models: these 'models' are used to derive a set of corporate plans, budgets and predicted financial statements. Many different algorithms may be employed (e.g. time-series, regression, simulation, subjective).

2. The corporate schema (T,B).

3. Data base for corporate and economic data.

4. Retrieval and display facilities.

(See Figure 2 above for an example of a typical CFM schema).

Although there are a multitude of FAM's in the management science literature, they normally consist of the following components:

1. Logical statement of model: e.g. a production, financial or marketing model using a variety of interconnected operations research algorithms and/or subjective estimates.

2. The subschema (T^S, B^S) where T^S is the set of transactions types and B^S the set of state variables which the FAM 'affects'.

3. FAM database.

4. Retrieval and display facilities.

5. Mapping functions: $(T^S, B^S) \rightarrow V^S$ and $V^S \rightarrow (T^S, B^S)$ where V^S is the set of all variables used in the model which are not elements of T or B.

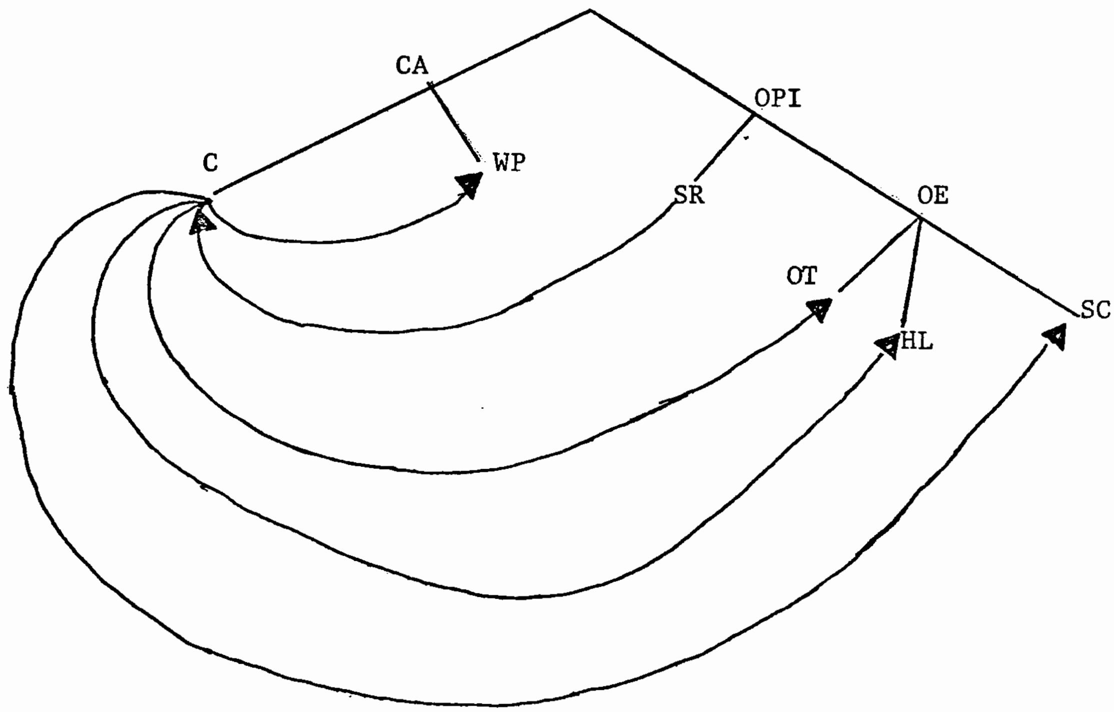
Note that there may be more than one FAM related to a given organizational function such as marketing and that the 'model' may consist of a number of related submodels employing different algorithms as long as these submodels are operated as a unit. As an example of a FAM consider the Holt, Modigliani, Muth and Simon (HMMS) aggregate workforce and inventory smoothing model [9]. The model variable set, V^S is:

p_t = aggregate production rate at time t

w_t = workforce level at time t

I_t = net inventory-on-hand at time t

The subschema and mapping are shown in Figure 4.



Transaction set, T^S

C/HL - Hiring and layoff costs
C/SC - Inventory back order and set up costs
C/OT - Expected overtime costs
C/WP - Regular payroll costs
SR/C - Sales Revenue (exogenous)

Mapping Rule, $V^S \rightarrow T^S$

$$C/HL = c_2(w_t - w_{t-1} - c_{11})^2 + c_{13}$$

$$C/SC = c_7(I_t - (c_8 - c_9S_t))^2$$

$$C/OT = c_3(p_t - c_4w_t)^2 + c_5p_t - c_6w_t + c_{12}p_tw_t$$

$$C/WP = c_1w_t + c_{13}$$

Figure 4

HMMS MODEL
(ONLY THOSE NODES AFFECTED BY
TRANSACTIONS IN THE MODEL ARE SHOWN)

The coefficients, c_i, are parameters to be estimated. Referring to Figure 4 it can be seen that, relative to the schema, some nodes in the subschema are aggregation points and some are disaggregation points representing a finer classification. Similarly, some transactions are aggregated and some disaggregated while the transaction SR/C occurs in both the schema and subschema.

The mappings $(T,B) \rightarrow (T^S,B^S)$ and $(T^S,B^S) \rightarrow (T,B)$ will require further rules to be stored by the planning system and these will depend on the accounting system chosen by the firm. One purpose of linking the FAM and CFM in this way is to assist the model-builder in estimating the parameters and exogenous variables required by the FAM. In the HMMS case past values of the cost transactions (at least C/WP and some aggregation of the others) might be made available for parameter estimation and forecast sales could be supplied whenever it is necessary to run the model.

Once one or several FAMs have been selected by the user their integration with the CFM through a top-down or bottom-up approach involves five major steps:

1. Input of Model Subschema (T^S,B^S).
2. Prereconciliation Stage (Mapping $(T,B) \rightarrow (T^S,B^S)$).
   The objective of the systems manager (SM) at this stage is to make the historical and projected time series data stored in the CFM compatible with the variables of the user's model. The model builder can then use the previous historic and projected data to estimate model parameters and examine historical relationships which may provide guidance in the specification of the model.
3. Model Construction.
   The MSG sub-system attempts to make the task of model specification simpler by automatically generating the matrices E and S

in equation (2) (which together contain all accounting identities) and providing other aids such as an integer, linear and goal programming tableau generator which works directly from an algebraic ('sigma notation') statement of the problem [12], [15].

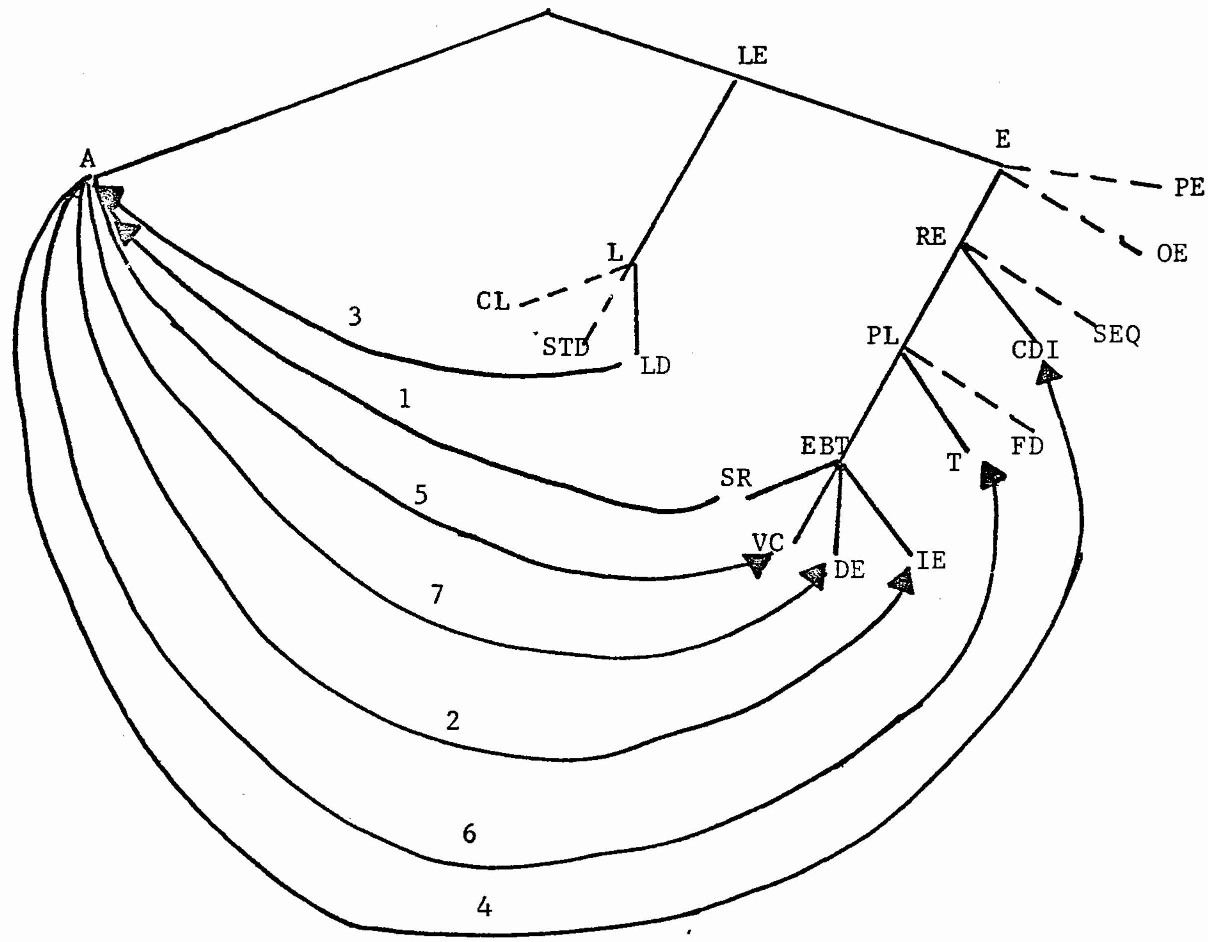
4. Running the Model.

   Data base techniques are used by the MM subsystem to manage the input and storage of data, model specification, the results of runs and the interrelationships between algorithms employed by the models [15].

5. Post-reconciliation Stage (Mapping $(T^S, B^S) \rightarrow (T, B)$).

   The objective of this stage is to translate the model results into a form compatible with the data stored in the CFM. This allows the user to examine corporate financial statements revised according to the results of his model. Since the model will be concerned only with a subset of the CFM transaction types the others have to be automatically estimated by the SM subsystem from the information stored in the CFM to complete the financial statements.

A detailed description of the pre- and post-reconciliation stages is given in [11] using a financial planning model proposed by Krouse [10] for illustration. As we shall use this model for our subsequent discussion of model representation, the Krouse model is now briefly described. The subschema and classification of variables is shown in Figure 5. The objective of the model is to determine optimal values for the debt acquired, d_t, the dividends paid, b_t, and the expense incurred, a_t, subject to achieving a set of financial goals--earnings growth and a number of balance sheet ratios (ratios of elements in B^S). A more detailed explanation of the Krouse model is given in Section V.



Transactions (Input by User)			Krouse Model	
No.	Affected Nodes (Cr/Db)	Description	Nature of Variable	Variable Definition
1	SR/A	Sales Revenue	exogenous	R
2	A/IE	Interest Expense	endogenous	$k_5 \cdot D$
3	LD/A	Debt Acquired	decision variable	b
4	A/CDI	Dividends Paid	decision variable	d
5	A/VC	Variable Expense	decision variable	a
6	A/T	Tax Expense	endogenous	$(1-k_6)P$
7	A/DE	Depreciation Expense	endogenous	

Figure 5

SUBSCHEMA FOR KROUSE MODEL

IV. A Canonical Representation for a Class of Corporate Planning Models

As mentioned earlier, we use the aggregate transaction vector τ to link the various FAM's and the CFM. However, certain major differences between variables must be kept in mind to understand the planning structure we propose. We recall first, that τ impacts on the state vector b in each period as stated by equation (2) for the model sub-schema (note that we will omit the 'S' superscript in this section):

(2) $$b^{t+1} = Eb^t + S\tau^t$$

The system stores definitions and historic and predicted values for the sub-schema-specific model variables in the state vector b^t and transaction vector τ^t. In addition, the user's model may contain other variables, v^t. A major purpose of the model support system is to automatically integrate the accounting identities contained in (2) with the behavioral equations of the model expressed in terms of b^t, τ^t and v^t. To achieve this we need the classification of variables shown in Figure 6. Here the suffices have the following meanings

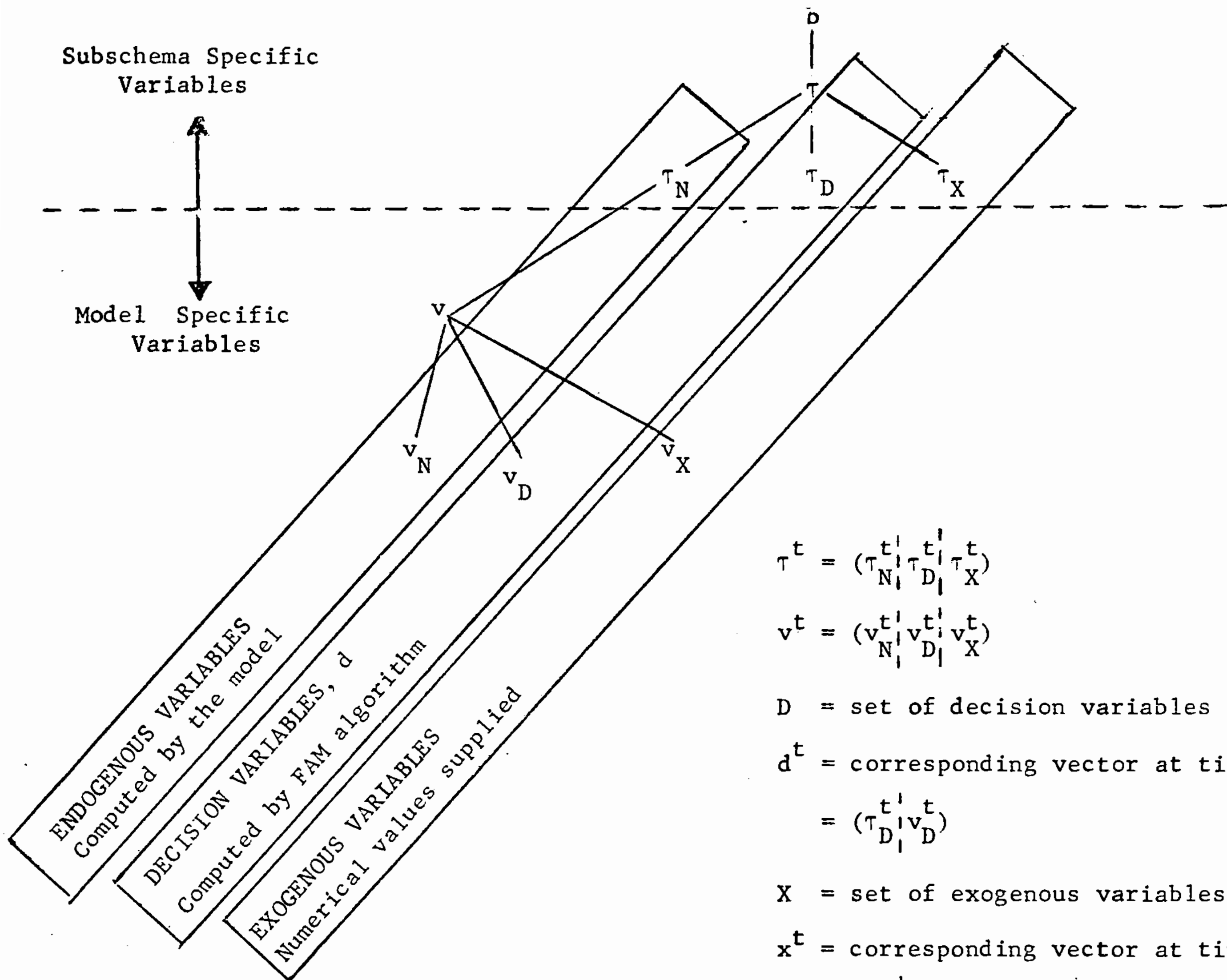
N - endogenous variables which are not decision (or directly controllable) variables

D - decision (or directly controllable) variables

X - exogenous variables

Note that all firm-specific monetary flows in the model are included in the transactor vector τ^t. In general, the variables in v^t will be non-monetary in nature, e.g. inventory levels expressed in units, aggregate workforce levels, etc. or will represent exogenous financial data such as GNP or total industry sales.

Figure 6 stresses the integration of the accounting identities logically contained in the sub-schema systems matrix, S, and equation (2) with the FAM model variables. Once this integration has been effected algebraically we will be able to express the evolution of the system recursively via a state equation of the type:



$\tau^t = (\tau_N^t \mid \tau_D^t \mid \tau_X^t)$

$v^t = (v_N^t \mid v_D^t \mid v_X^t)$

D = set of decision variables

d^t = corresponding vector at time t

$= (\tau_D^t \mid v_D^t)$

X = set of exogenous variables

x^t = corresponding vector at time t

$= (\tau_X^t \mid v_X^t)$

Figure 6

CLASSIFICATION OF MODEL VARIABLES

$$(3) \qquad \tilde{b}^{t+1} = \tilde{A}_B \tilde{b}^{t} + \tilde{A}_D d^{t} + \sum_{s=0} \tilde{A}^{s} x^{t-s}$$

where $\tilde{b}^t = (b^t \mid v_N^t \mid \tau_N^{t-1})$ and $\tilde{A}_B, \tilde{A}_D, \tilde{A}_X^s$ are conformable matrices.

Depending on the solution algorithm used (e.g. linear-quadratic control model, linear programming or goal programming) the model builder will be required to input also a set of goals (or targets) and a set of constraints on the values of variables. These will have the form:

Goals:

$$(4) \qquad g^{t} = \tilde{G}_B^{t+1} \tilde{b}^{t+1} + \tilde{G}_B^{t} \tilde{b}^{t} + \tilde{G}_D^{t} d^{t} + \sum_{s=0} \tilde{G}_X^{s} x^{t-s}$$

where g^t is a vector of target values and $\tilde{G}_B^{t+1}$, $\tilde{G}_B^t$, $\tilde{G}_D^t$, $\tilde{G}_X^t$, are matrices of constants embodying the user's goal specifications.

Constraints:

$$(5) \qquad r^{t} \substack{\leq \\ = \\ \geq} \tilde{R}_B^{t+1} \tilde{b}^{t+1} + \tilde{R}_B^{t} \tilde{b}^{t} + \tilde{R}_D^{t} d^{t} + \sum_{s=0} \tilde{R}_X^{s} x^{t-s}$$

where r^t is a vector of constants and $\tilde{R}_B^{t+1}$, $\tilde{R}_B^t$, $\tilde{R}_D^t$, $\tilde{R}_X^s$, are matrices of constants embodying the constraint conditions.

Equations (3), (4) and (5) are developed internally by the computer system by interpreting statements in the planning language. Together they constitute a standard representation for a wide class of planning models. When a particular algorithm is chosen by the user an interface system associated with the algorithm transforms the standard representation to the exact format required by the algorithm. For example, if goal programming is used, equation (4) will represent the goal equations and equations (3) and (5) the constraints .

Finally, the source of values for the variables used in our model must be clearly understood:

(i) τ_D and v_D are computed by whatever algorithm is used in the FAM adopted by the planner. Hence they require no behavioral equations.

(ii) τ_X can be: - supplied automatically by the SM subsystem (see [11])

or: - retrieved from external files

or: - stated in user-supplied equations as functions of v_X

(iii) v_X can be: - retrieved from external files

or: - stated in terms of τ_X

(by definition the exogenous variables, $x = (\tau_X, v_X)$, cannot be functions of any other variables of the model).

(iv) τ_N and v_N, being endogenous are computed from user-supplied equations.

We now return to our stated goal of deriving a canonical representation of a class of planning models--i.e. equations (3), (4) and (5). An analogy with the concept of structured programming [16], may help the reader understand the rationale for our approach. In structured programming, programmers are constrained to use a subset of language 'structures' which enforce a simplified, modular logical statement of the problem. At the same time a high degree of uniformity in programming style is achieved. Existing OR-type planning models combine in various degrees behavioral equations, accounting identities and definitional equations. As their breadth and depth of variable coverage varies, no easy comparison can be made between them. More importantly it is not possible for the planner to take one model and have it translated automatically into another. Any new model considered by the planner must be made consistent with other existing models if outputs are to be compared. And finally and most importantly, it is never clear to the planner how he should build his own model; what minimal building blocks he needs and how they fit together. The planner is literally faced with a jig-saw puzzle which he must assemble by relying on his understanding of corporate functional areas and accounting. This is undesirable if planning is to be undertaken frequently and systematically. By contrast, our approach is structured and model building is guided throughout to help the user:

1. Notation and all subschema specific model variables and definitions are predefined.
2. Users are required to use a common classification scheme for variables as described above.
3. Model builders are forced to use a transaction-based viewpoint when building models.

We now show how the state equation (3) describing the evolution of the system is derived during step 3 of the model building process (see p. 15). We note first the user-supplied expressions for τ_N^t and v_N^t as linear functions of current and lagged values of the states, the decisions, the exogenous variables, and other endogenous variables namely:

$$
\begin{aligned}
(6) \qquad v_N^{t+1} = {} & M_B^{(-1)} b^{t+1} + M_B^{(0)} b^t + M_B^{(1)} b^{t-1} + \dots \\
& + M_D^{(0)} d^t + M_D^{(1)} d^{t-1} + \dots \\
& + M_X^{(0)} x^t + M_X^{(1)} x^{t-1} + \dots \\
& + M_T^{(0)} \tau_N^t + M_T^{(1)} \tau_N^{t-1} + \dots \\
& + M_V^{(-1)} v_N^{t+1} + M_V^{(0)} v_N^t + M_V^{(1)} v_N^{t-1} + \dots
\end{aligned}
$$

$$(7)\quad \tau_N^t = L_B^{(-1)}\, b^{t+1} + L_B^{(0)}\, b^t + L_B^{(1)}\, b^{t-1} + \ldots$$
$$+ L_D^{(0)}\, d^t + L_D^{(1)}\, d^{t-1} + \ldots$$
$$+ L_X^{0}\, x^t + L_X^{(1)}\, x^{t-1} + \ldots$$
$$+ L_V^{(-1)}\, v_N^{t+1} + L_V^{(0)}\, v_N^t + L_V^{(1)}\, v_N^{t-1} + \ldots$$
$$+ L_T^{(1)}\, \tau_N^{t-1} + \ldots$$

where, for example, the notation $L_B^{(s)}$ denotes the constant matrix multiplier for the vector of state variables at time t-s. Although not explicit in (6) and (7) if a particular component of τ_N^t is expressed in terms of a particular component of v_N^t then this component of v_N^t will not in turn be expressed as a function of that component of τ_N^t. This avoids circularity in the definitions.

Partitioning the matrices in (2), (6) and (7) according to the classification described earlier we obtain general expressions for the accounting system state equation and the user-defined behavioral equations:

$$(2')\quad b^{t+1} = Eb^t + S_D\tau_D^t + S_N\tau_N^t + S_X\tau_X^t \qquad , \; t=0,1,2,\ldots$$

(6′)

$$v_N^{t+1} = M_B^{(-1)} b^{t+1} + M_B^{(0)}\, b^t + \sum_{s=1} M_B^{(s)}\, b^{t-s}$$
$$+ [M_{DT}^{(0)} \mid M_{DV}^{(0)}] d^t + \sum_{s=1} [M_{DT}^{(s)} \mid M_{DV}^{(s)}] d^{t-s}$$
$$+ [M_{XT}^{(0)} \mid M_{XV}^{(0)}] x^t + \sum_{s=1} [M_{XT}^{(s)} \mid M_{XV}^{(s)}] x^{t-s}$$
$$+ M_T^{(0)}\, \tau_N^t + \sum_{s=1} M_T^{(s)}\, \tau_N^{t-s}$$
$$+ M_V^{(-1)} v_N^{t+1} + M_V^{(0)}\, v_N^t + \sum_{s=1} M_V^{(s)}\, v_N^{t-s} \quad , \; t=0,1,2,\ldots$$

$$(7') \quad \tau_N^t = L_B^{(-1)} b^{t+1} + L_B^{(0)} b^t + \sum_{s=1} L_B^{(s)} b^{t-s}$$
$$+ [L_{DT}^{(0)} \mid L_{DV}^{0}] d^t + \sum_{s=1} [L_{DT}^{(s)} \mid L_{DV}^{(s)}] d^{t-s}$$
$$+ [L_{XT}^{(0)} \mid L_{XV}^{(0)}] x_t + \sum_{s=1} [L_{XT}^{(s)} \mid L_{XV}^{(s)}] x^{t-s}$$
$$+ L_V^{(-1)} v_N^{t+1} + L_V^{(0)} v_N^t + \sum_{s=1} L_V^{(s)} \tau_N^{t-s}$$
$$+ L_T^{(0)} \tau_N^t \qquad , t = 0,1,2,\ldots$$

Equations (6′) and (7′) are the most comprehensive representation of the user definition of a physically realizable linear system. Taking equations (6′) and (7′), we note that they describe the model specific vectors of variables in terms of (1) accounting states (b), (2) decision variables (d), (3) exogenous variables (x), (4) endogenous transactions (τ_N), and (5) values of the endogenous model specific variables (v_N). We further note that b and v_N may contain stock variables in terms of which other components of these vectors may be expressed. For instance, the tax and interest expense transactions (#23 and #24, Figure 2) can only be determined at the end of the accounting period. Thus τ_N^t must be expressed in terms of b^{t+1}. Also note the treatment of time in these equations. First, b and v_N are both expressed at time (t+1) on the left-hand side of these equations because they contain some stock variables (however, note that it is also convenient to include flow variables in b and v_N). On the other hand, τ_N^t only contains flow variables at time t which yield the values of the stock variables at time t+1. We further notice that equation (2′) contains only variables in t on the right-hand side whereas equations (6′) and (7′) contain variables in (t+1). This does not make (2′) any less general since τ_N^t is expressed in terms of b^{t+1}. The terms $M_{(V)}^{(1)} v_N^{t+1}$ in (6′) and $L_T^{(0)} \tau_N^t$ in (7′) allow the model builder (for example) to express some endogenous variables as constant ratios of other endogenous variables in the same time period. Finally lagged variables from periods prior to t are included to allow current values to be affected by any combination of past values. This flexibility is required, for instance, in models of accounts receivable.

Taking equations (2′), (6′) and (7′), the planning system can now express the state equation of the system in terms of a state vector

$\tilde{b}^t = \begin{pmatrix} b^t \\ v_N^t \\ \tau_N^{t-1} \end{pmatrix}$. This is given by:

$$(8) \qquad A_B^{(-1)} \tilde{b}^{t+1} = A_B^{(0)} \tilde{b}^t + A_D^{(0)} d^t + \sum_{s=0} A_x^{(s)} x^{t-s} \qquad , t = 0,1,2,\ldots$$

where

$$A_B^{(-1)} = \left[\begin{array}{c|c|c} I & 0 & -S_N \\ \hline M_B^{(-1)} & I-M_V^{(-1)} & -M_T^{(0)} \\ \hline -L_B^{(-1)} & -L_V^{(-1)} & I-L_T^{(0)} \end{array}\right]$$

$$A_B^{(0)} = \left[\begin{array}{c|c|c} E & 0 & 0 \\ \hline M_B^{(0)} & M_V^{(0)} & M_T^{(1)} \\ \hline L_B^{(0)} & L_V^{(0)} & L_T^{(1)} \end{array}\right]$$

$$A_D^{(0)} = \left[\begin{array}{c|c} S_D & 0 \\ \hline M_{DT}^{(0)} & M_{DV}^{(0)} \\ \hline L_{DT}^{(0)} & L_{DV}^{(0)} \end{array}\right]$$

$$A_X^{(s)} = \left[\begin{array}{c|c} S_X^{(s)} & 0 \\ \hline M_{XT}^{(s)} & M_{XV}^{(s)} \\ \hline L_{XT}^{(s)} & L_{XV}^{(s)} \end{array}\right]$$

Note that to simplify the notation we have eliminated terms in d^{t-s}, v_N^{t-s} and b^{t-s} for $s > 0$ and terms in τ_N^{t-s} for $s > 1$. Including these terms would increase the dimension of the state space representation and complicate the exposition without making any fundamental difference in the theory. Premultiplying both sides of equation (8) by $[A_B^{(-1)}]^{-1}$ yields equation (3). Clearly this inverse will exist for any well-specified model. Finally the matrices in equations (4) and inequalities (5) can be similarly derived.

V. Example - Specification of a Planning Model Based on Control Theory

To illustrate the specification of a model using the standard representation discussed in the previous section we use the illustrative model developed by Krouse [10] which was introduced earlier. This was chosen because it typifies a number of high level financial planning models while containing relatively few variables. We first introduce the model using the original notation and then show how it would be input using the notation and procedures of the Model Statement Generator (MSG) subsystem of the planning system. Figure 7 shows this model in the 'structural' form given in [10] omitting the additive random disturbance terms.

Original Notation		Planning System Notation
(9-a)	$R(t+1) = k_1R(t) + k_2A(t+1) + k_3C(t+1)$	$b^{t+1}(SR/A) = k_1b^t(SR) + k_2b^{t+1}(A) + k_3C_{t+1}$
(9-b)	$P(t+1) = R(t+1) - a(t) - k_4A(t+1) - k_5D(t+1)$	$b^{t+1}(EBT) = \tau^t(SR/A) - \tau^t(A/VC) - k_4b^{t+1}(A) - k_5b^{t+1}(LD)$
(9-c)	$A(t+1) = (1-k_4)A_t + k_6P(t) + b_t - d_t$	$b^{t+1}(A) = (1-k_4)(b^t(A) + k_6b^t(EBT) + \tau^t(LD/A) - \tau^t(A/CDI)$
(9-d)	$D_{t+1} = D_t + b_t$	$b^{t+1}(LD) = b^t(LD) + \tau^t(LD/A)$
(9-e)	$E(t+1) = E(t) + k_6P(t) - d(t)$	$b^{t+1}(E) = b^t(E) + k_6b^{t+1}(EBT) - \tau^t(A/CDI)$
(9-f)	$C(t+1) = k_7C(t) + (1-k_7)a(t)$	$C(t+1) = k_7C(t) + (1-k_7)\tau^t(A/VC)$
(9-g)	$F(t+1) = E(t+1) - 2D(t+1)$	$F(t+1) = b^{t+1}(E) - 2b^{t+1}(LD)$
(9-h)	$Y(t+1) = R(t+1) - 7P(t+1)$	$Y(t+1) = b^{t+1}(SR) - 7b^{t+1}(EBT)$
(9-i)	$H(t+1) = P(t+1) - 4d(t)$	$H(t+1) = b^{t+1}(EBT) - 4\tau^t(A/CDI)$

State Variables

R = Revenues
A = Assets
C = "Effective" Cost Level
D = Cumulating Debt
E = Equity

P = Profits
F = Debt/Equity Ratio
Y = Profit/Revenue Ratio
H = Divident Payout Ratio

Decision Variables

a = Period variable expenses
b = New debt incurred or retired
d = Dividends paid

Constants

k_1, k_2, k_3 Regressors of the revenue behavioral equa
k_4 Overall depreciation rate on assets
k_5 Rate of interest on debt
k_6 One minus the tax rate
k_7 Smoothing constant

Figure 7

FINANCIAL PLANNING MODEL - ORIGINAL NOTATION AND FORMAT

The figure also shows the 'planning system' notation for these variables. This will be explained later. Equation (9-a) expresses revenues as a function (derived by regression procedures) of previous revenues R(t), assets A(t+1), and 'effective expenditures', C(t+1). Equation (9-b) relates earnings before taxes, P(t+1) to revenues, R(t+1), variable production and promotion expenses, a(t), assets A(t+1), and the level of debt, D(t+1). Equations (9-c), (9-d) and (9-e) are accounting identities combined (in the case of equations (9-c) and (9-e)) with behavioral assumptions concerning depreciation, interest and tax rates. Equation (9-f) defines 'effective' expenditures C(t+1) as an exponentially smoothed series with a carry-over effect of expenditures from past periods. Equations (9-g), (9-h) and (9-i) define management's target debt/equity, profit/revenue and dividend payout ratios.

Target values for F(t), Y(t) and H(t) and for profits, P(t), in each time period are supplied by management. The objective function is a weighted quadratic function of the squared deviations of F(t), Y(t), H(t) and P(t) from their respective target values. The model is solved in each period using the linear-quadratic control theory algorithm (see, for example, Chow [5] to yield an optimal decision rule which is linear in the decision variables a(t), b(t), d(t).

To define this model using the framework discussed in this paper the model subschema is first defined as described in Stohr and Tanniru [11]. The graphical representation of this subschema is given in Figure 5 above. Note that the subschema does not correspond exactly to the schema in Figure 2. In particular the sub-tree below EBT has been rearranged and a new code, VC (Variable Costs) has been introduced. The manner in which the systems manager subsystem keeps track of these changes during the 'Preconciliation Stage' is described in [11]. It should be noted, however, that after the subschema has been entered (by defining the tree and transaction arcs) the planning system is capable of retrieving historic values of all node and transaction values including those for the node VC and transaction A/VC. In addition, past balance sheet, income statement and cash flow statements especially tailored to the model can be produced in both tabular and graphical form. This capability can help give the model builder a better 'feel' for the relationships in the model and can, of course, be used directly to produce regression results such as those required by the Krouse model. The notation which will be used in this paper to refer to the states and transactions of the subschema is indicated by the following two examples: (1) the state value for SR (sales revenue) at time t is expressed as B(SR,T) in the planning language but will be referred to as $b^t(SR)$ in this paper; (2) the value of the transaction C/IE (interest expense) at time t is expressed as T(C/IE,T) in the planning language but will be referred to as $\tau^t(C/IE)$ in this paper. Figure 7 above shows the correspondence between the original notation used in [10] and the planning system notation. However it must be emphasized that the structured approach to modelling adopted here requires that the model be stated in a different (although equivalent) manner. To avoid ambiguity a convention is introduced in the planning language to handle statements involving nodes in the subschema tree that correspond to flow variables (i.e. income statement nodes). Note that the matrix E in (2) causes the previous time periods' values of these nodes to be zeroed so that the value of the node at time period t+1 equals the net transaction flow through the node. Since only one subschema transaction is associated with the leaf nodes SR, VC, DE, IE, T, CDI in Figure 5 the node and associated transaction values are equal--thus $b^{t+1}(SR) = \tau^t(SR/A)$, $b^{t+1}(VC) = \tau^t(A/VC)$, etc. However, for example, $b^{t+1}(EBT) = \tau^t(SR/A) + \tau^t(A/VC) + \tau^t(A/DE) + \tau^t(A/IE)$. Also note that the direction of transaction flow accounts for the sign so that the expense transactions in the previous formula are added to the revenue transaction rather than subtracted.

Figure 8 shows the interaction of the model builder with the MSG subsystem during the model definition phase. Prompts from the computer are underlined and statements beginning with asterisks are comments. The result of the model definition is an APL character matrix which (after editing by the user) is interpreted to produce the standard model representation discussed in the previous section.

In step 1 of the model building process the user classifies the model variables as decision (or controllable) variables, other endogenous variables, exogenous variables and parameters.

In step 2 the planning system language (see [13]) is used to define values for the parameters and initial values for lagged model specific variables. Values for the initial values of lagged subschema variables are automatically supplied by default. Note that T, the time index variable is a special variable in the planning language and that the statement, 'P[T] IS 30 GROW 10', specifies that the target growth rate for profits is 10% per time period.

In step 3 the user is prompted to supply expressions in the form implicit in equations (6) and (7) for the variables τ_N^t and v_N^t. In step 4 the goals are defined and in step 5 the constraints (there are none in this model). If this model contained exogenous variables, the user would have been prompted for their definitions in a manner similar to step 3.

The model statements in steps 3, 4 and 5 are interpreted by a modified form of the mathematical programming tableau generator system described in [9]. Note that this system accepts any valid APL statement as a variable coefficient--for example, the coefficient (1 - K[7]) is evaluated by the APL host language.

1. VARIABLE DECLARATIONS:

DECISION VARIABLES:

T(LD/A,T),T(A/CDI,T),T(A/VC,T)

OTHER ENDOGENOUS VARIABLES:

T(SR/A,T),T(A/IE,T),T(A/T,T),T(A/DE,T),C(T)

EXOGENOUS VARIABLES:

PARAMETERS:

K1 thru K7, F[T], Y[T], H[T], P[T]

2. INPUT PARAMETER VALUES AND INITIAL VALUES FOR LAGGED VARIABLES IF ANY:

K IS .5 .1 .65 .2 .08 .5 .4

C[0] IS 160

F[T] IS 0

Y[T] IS 0

H[T] IS 0

P[T] IS 30 GROW 10

3. INPUT DEFINITIONS FOR ENDOGENOUS VARIABLES:

(1) T(SR/A,T) - SALES REVENUE:

K[1] T(SR/A,T-1) + K[2] B(A,T+1) + K[3] C(T+1)

(2) T(A/IE,T) - INTEREST EXPENSE:

K[5] B(LD,T+1)

(3) T(A/T,T) - TAX EXPENSE:

(1 - K[6]) B(EBT,T+1)

(4) T(A/DE,T) - DEPRECIATION EXPENSE:

K[4] B(A,T)

(5) C(T+1) - USER DEFINED VARIABLE:

K[7] C(T) + (1 - K[7]) T(A/VC,T)

4. INPUT DEFINITIONS FOR GOALS:

* DEBT-EQUITY RATIO

B(E,T) - 2 B(LD,T) = F(T)

* SALES TO PROFIT RATIO

B(SR,T) - 7 B(PL,T) = Y(T)

* PROFIT TO DIVIDEND RATIO

B(PL,T) - 4 B(CDI,T) = H(T)

* PROFIT GOAL

B(PL,T) = P(T)

5. INPUT DEFINITIONS FOR CONSTRAINTS:

Figure 8

DEFINITION OF A FINANCIAL PLANNING MODEL

Comparing Figures 8 and 9 we see the changes in the model statement which result from the structured transaction-oriented modelling approach advocated in this paper. The model builder defines expressions for the variables τ^t_N, v^t_N, τ^t_X, and v^t_X but all accounting identities are supplied by the planning system. Thus equations (10-b) through (10-e) which consist of a mixture of accounting relationships and behavioral assumptions are replaced by a set of expressions (Step 3, (2) through (4)), which contain behavioral equations for individual transactions.

After the statement of the model by the user as shown in Figure 8, the MSG subsystem will perform consisting checks to insure that all variables can be recognized and that the behavioral equations conform to the model given by equations (6) and (7). It then produces the standard representation, (8), of the model. For the Krouse model this reduces to: $A^{(-1)}_B b^{t+1} = A^{(0)}_B b^t + A^{(0)}_D d^t$.

These matrices are shown in Figure 9 for the Krouse model and the final 'reduced' form (equation 3) in Figure 10. (Note that the parameter definitions are modified somewhat in the transaction-oriented approach adopted here.)

Automatically imbedding this operations research model within the accounting framework in this way yields much useful information concerning the sensitivity of subschema financial variables (some of which were not included in the original statement of the model) to changes in the model decision variables. Thus, from Figure 8 we see that a one dollar increase in long-term debt translates one period later to a \$1.01 increase in assets.

The HMMS model described earlier will have a similar standard representation. In this case, as can be seen from Figure 4, most of the subschema transactions do not appear directly in the model. However, since they are terms in the model's cost function, they can be computed once the optimal

$A_B^{(-1)} =$

	A	LE	L	E	LD	RE	PL	CDI	EBT	T	SR	VC	DE	IE	C	SR/A	A/IE	A/T	A/DE
A	1															-1	1	1	1
LE		1														1	-1	-1	-1
L			1													0	0	0	0
E				1												1	-1	-1	-1
LD					1						0					0	0	0	0
RE						1										1	-1	-1	-1
PL							1									1	-1	-1	-1
CDI								1							0	0	0	0	0
EBT									1							1	-1	0	-1
T					0					1						0	0	-1	0
SR											1					1	0	0	0
VC												1				0	0	0	0
DE													1			0	0	0	-1
IE														1		0	-1	0	0
C							0								1				
SR/A	$-k_2$														$-k_3$	1			0
A/IE					$-k_5$												1		
A/T									$-(1-k_6)$						0			1	
A/DE															0	0			1

, $A_B^{(0)} =$

	A	LE	L	E	LD	RE	PL	CDI	EBT	T	SR	VC	DE	IE	C	SR/A	A/IE	A/I	A/DE
A	1																		
LE		1																	
L			1																
E				1						0									
LD					1														
RE						1													
PL							0								0			0	
CDI								0											
EBT									0										
T				0						0									
SR											0								
VC												0							
DE													0						
IE														0					
C							0								k_7			0	
SR/A																k_1			
A/IE							0												
A/T															0				
A/DE	k_4																		

, $A_D^{(0)} =$

	LD/A	A/CDI	A/VC
A	1	-1	-1
LE	-1	1	1
L	-1	0	0
E	0	1	1
LD	-1	0	0
RE	0	1	1
PL	0	0	1
CDI	0	1	0
EBT	0	0	1
T	0	0	0
SR	0	0	0
VC	0	0	1
DE	0	0	0
IE	0	0	0
C	0	0	$1-k_7$
SR/A			
A/IE			
A/T		0	
A/DE			

Figure 9

THE FINANCIAL PLANNING MODEL IN STRUCTURAL FORM

	A	LE	L	E	LD	RE		C	SR/A	A/IE	A/T	A/DE		LD/A	A/CDI	A/VC	
A	0.95	0.00	0.00	0.00	0.04	0.00		0.14	0.26	0.00	0.00	0.00		1.01	-1.05	-0.32	
LE	0.05	1.00	0.00	0.00	-0.04	0.00		-0.14	-0.26	0.00	0.00	0.00		-1.01	1.05	0.32	
L	0.00	0.00	1.00	0.00	0.00	0.00		0.00	0.00	0.00	0.00	0.00		-1.00	0.00	0.00	
E	0.05	0.00	0.00	1.00	-0.04	0.00		-0.14	-0.26	0.00	0.00	0.00		-0.01	1.05	0.32	
LD	0.00	0.00	0.00	0.00	1.00	0.00		0.00	0.00	0.00	0.00	0.00		-1.00	0.00	0.00	
RE	0.05	0.00	0.00	0.00	-0.04	1.00		-0.14	-0.26	0.00	0.00	0.00		-0.01	1.05	0.32	
PL	0.05	0.00	0.00	0.00	-0.04	0.00		-0.14	-0.26	0.00	0.00	0.00		-0.01	0.05	0.32	
CDT	0.00	0.00	0.00	0.00	0.00	0.00		0.00	0.00	0.00	0.00	0.00		0.00	1.00	0.00	
EBT	0.11	0.00	0.00	0.00	-0.08	0.00		-0.27	-0.53	0.00	0.00	0.00		-0.02	0.11	0.64	
$\tilde{b}_t$ = T	-0.05	0.00	0.00	0.00	0.04	0.00	0	0.14	0.26	0.00	0.00	0.00	$\tilde{b}_t$ +	0.01	-0.05	-0.32	d^t
SR	-0.09	0.00	0.00	0.00	0.00	0.00		-0.27	-0.53	0.00	0.00	0.00		-0.10	0.11	-0.36	
VC	0.00	0.00	0.00	0.00	0.00	0.00		0.00	0.00	0.00	0.00	0.00		0.00	0.00	1.00	
DE	0.20	0.00	0.00	0.00	0.00	0.00		0.00	0.00	0.00	0.00	0.00		0.00	0.00	0.00	
IE	0.00	0.00	0.00	0.00	-0.08	0.00		0.00	0.00	0.00	0.00	0.00		0.08	0.00	0.00	
C	0.00	0.00	0.00	0.00	0.00	0.00		0.40	0.00	0.00	0.00	0.00		0.00	0.00	0.60	
SR/A	0.09	0.00	0.00	0.00	0.00	0.00		0.27	0.53	0.00	0.00	0.00		0.10	-0.11	0.36	
A/IE	0.00	0.00	0.00	0.00	-0.08	0.00		0.00	0.00	0.00	0.00	0.00		0.08	0.00	0.00	
A/T	-0.05	0.00	0.00	0.00	0.04	0.00		0.14	0.26	0.00	0.00	0.00		0.01	-0.05	-0.32	
A/DE	0.20	0.00	0.00	0.00	0.00	0.00		0.00	0.00	0.00	0.00	0.00		0.00	0.00	0.00	

Figure 10

THE FINANCIAL PLANNING MODEL IN REDUCED FORM values of the variables, V^s, have been obtained by the algorithm. Thus the post-reconciliation step of the modeling process will employ the mapping $V^s \rightarrow T^s$ given in the Figure.

We will not describe the next phase (step 4: Model Running) of the planning system in detail here. However, it must be emphasized again that the standard representation shown in Figure 8 for the Krouse model can be used as input to a number of different algorithms:

(1) Linear programming: equation (8) (see Figure 10) can be used to generate a single time period or multiple time period tableau of constraints. The model objectives in the expression of the goals (Figure 8, step 4) can be incorporated in the objective function (as a weighted sum of absolute deviations from the goals).

(2) Goal-programming: the model constraints are again determined from (8) and the goal equations from the expressions in step 4 of Figure 8. The model-builder must only add the priorities and their weights for the definition of the multiple criterion objective function.

(3) Linear quadratic Gaussian control theory algorithms: the linear systems equation is given by (8). The quadratic objection function, which minimizes the weighted sum of the squared deviations of model variables from the ideal trajectories given by the goal definitions in Figure 8, must be supplied by the user.

(4) Simulation: the linear systems equation (8) can be used to generate directly the time-paths of all financial and model variables for any trial settings of the model decision variables. These trajectories can be displayed graphically or in the form of standard financial reports. In addition,criterion functions can be defined by the model-user to help in evaluating the different trial policies. Note that the simula-

tion mode might be entered after 'optimal' values for the decision variables have been set by one of the operations research algorithms. Other measures of performance, sensitivity tests and managerial insights could then be used to adjust the 'optimal' decisions in order to determine a final course of action for the firm. Note also that the model defined by Figure 10 could easily be expanded by additional statements (perhaps involving logical conditions) in the planning language. This could be used to expand the original optimization model to cover other aspects important to the corporation or to add refinements to the model which were not computationally feasible using the optimization technique.

The transformations from the standard representation which are required to run each of the above algorithms are performed by the MM subsystem with guidance from the user.

References

1. Alter, Steven, "A Taxonomy of Decision Support Systems," Sloan Management Review, Fall 1977, pp. 39-56.

2. Blin, J.M., E.A. Stohr and M. Tanniru, "Design of a Corporate Information System," Proceedings IEEE COMPSAC Conference, Chicago, November 1977, pp. 700-710.

3. Bonczek, R., C. Holsapple and A. Whinston, "A Theoretical Design for a Decision Support System," Proceedings IEEE COMPSAC '77, Chicago, November 1977, pp. 744-750.

4. CALL/370: PROPHIT II Reference Manual, The Service Bureau Company, February 1977.

5. Chow, Gregory, Analysis and Control of Dynamic Economic Systems, John Wiley and Sons, New York, New York, 1975.

6. FPS User's Guide, Common Library 701, Scientific Time Sharing Corporation, December 1973.

7. Gorry, G.A. and M.S. Scott-Morton, "A Framework for Management Information Systems," Sloan Management Review, Vol. 13, No. 1, 1971, pp. 55-70.

8. Haseman, William D., "Framework for a Planning Information System," Ph.D. Dissertation, Purdue University, May 1975.

9. Holt, C.C., F. Modigliani, J.F. Muth and H.A. Simon, Planning Production, Inventories and Workforce, Prentice-Hall, Englewood Cliffs, New Jersey, 1960.

10. Krouse, Clement G., "A Model for Aggregate Financial Planning," Management Science, Vol. 18, No. 10, June 1972, pp. B555-B566.

11. Stohr, E.A. and M. Tanniru, "Design of a Corporate Planning System Simulator," Proceedings ACM Winter Simulation Conference, Miami, December 1978.

12. Stohr, E.A., "A Mathematical Programming Generator System in APL," Working Paper No. 348, Center for Mathematical Studies in Economics, Northwestern University, September 1978.

13. Stohr, E.A., "CORPLAN User's Manual," (unpublished manuscript).

14. Subas, S.K.C. and R.L. Kashyap, "Data Base Organization--An Access and Storage Method," Ph.D. Dissertation, School of Electrical Engineering, Purdue University, September 1975.

15. Tanniru, Mohan, "Design of a Decision Support System," Ph.D. Dissertation, Graduate School of Management, Northwestern University, 1978.

16. Yourdon, Edward, <u>Techniques of Program Structure and Design</u>, Prentice- Hall, Englewood Cliffs, New Jersey, 1975.