Representing and Reasoning About Costs Using Enterprise Models and ABC

By

Kokchu Donald Tham

A Thesis submitted in conformity with the requirements
for the degree of Doctor of Philosophy
Graduate Department of Mechanical and Industrial Engineering
University of Toronto

Representing and Reasoning About Costs Using Enterprise Models and ABC

Doctor of Philosophy (1999)

Kokchu Donald Tham

Department of Mechanical and Industrial Engineering

University of Toronto

Abstract

The well accepted Activity-Based Costing (ABC) Principle includes the assignment of costs to activities based on their use of resources, and the assignment of costs to end products and services based on their use of activities. Since ABC assigns costs to activities based on their use of resources, the logical formulation of ABC must be premised upon the existence of some given or identifiable unit resource cost that must be associated with each resource required by an activity. Notwithstanding this obvious rationale for unit resource cost being absolutely essential for ABC, there has been little or no development in the field of ABC to solve two fundamental problems relevant to ABC:- *(i) What unit resource costs are associated with a resource? (ii) How does one deduce unit resource costs so that direct, indirect and overhead costs are accounted for within the costs of a resource?* The solving of these two fundamental problems relevant to ABC is the central focus of this research.

Though centered within the cost domain, this research presents a holistic solution by crossing over several fields which, in themselves, are expansive and highly specialized areas. More specifically, the perspectives integrated are from the fields of:- (a) Computer Science and Artificial Intelligence for the development of knowledge-based information systems and the deductive reasoning of solutions in a distributed agent environment; (b) "New Industrial Engineering" that applies the capabilities of information technology to analyze, engineer and re-engineer business processes on an enterprise wide basis;

(c) Strategic Cost Management that seeks accurate and consistent solutions for product and service costs to formulate strategies for competitive advantages.

First, based upon real cost related problems put forth as common sense queries posed by industrial partners, this research presents the analysis and design of a Cost Ontology for Enterprise Modelling that identifies and reasons with Resource Cost Units of a resource. A Formalization of ABC and a Micro-theory of ABC are established based upon the Cost Ontology. Secondly, it develops the Principle of Resource Probing, the Principle of Activity Probing, a Formalization of Overheads, and a Micro-theory of Resource Cost Units that collectively form a Theory of Resource Cost Units. This theory establishes the framework of axioms and concepts that enables one to deduce costs with ABC so that direct, indirect and overhead costs are included in the Resource Cost Units.

The theories and developments of this research are tested for practical applicability by finding solutions to real world scenarios as presented by corporate partners - deHavilland Inc. (Bombardier, Canada) and BHP Steel (Broken Hill Proprietary, Australia) for purposes of strategic cost management.

Dedication

This research is dedicated to my dearest wife, Shuyee, our loveable twins, Jason and Karen, my caring parents, and my late elder brother, Monchu, who passed away suddenly while this endeavour was in progress.

Acknowledgments

Effective adult learning results from real practical experiences and being amongst individuals who have positive attitudes, strive to achieve, willingly share their experiences, accept and objectively criticize one's point of view for one's betterment, and, above all, accept you in the spirit of a trusting colleague. This has been the nature of my enviable and enjoyable experience from the day I embarked upon my doctoral studies within the Enterprise Integration Laboratory (EIL) of Professor Mark Fox. Being physically located within the Center of Management of Technology and Entrepreneurship (CMTE) of Professor Joe Paradi, I was thus fortunate of being amidst the best of two worlds - EIL and CMTE.

I am grateful to Mark Fox, my thesis supervisor, for all the assistance, guidance, and the very challenging and insightful queries brought forth that improved the quality of this research. To my committee members, Joe Paradi, Anthony Wensley (Mgmt. Faculty, U of T) and Murray Bryant (External Examiner, Ivey Business School) - thanks for your critique, suggestions and for being so approachable. Thanks to the deHavilland and BHP companies for their support and cooperation so that I could demonstrate this research as being practical and relevant to strategic cost management.

To my partner in the doctoral stream, Henry Kim: many thanks for our "varied discussions", which, I am sure, contributed to "ISO - certify" my work. Many thanks to members (past and present) of EIL - John Chionglo, Fadi Fadel, Yin Zhan, Wai-Kau Lo, Mihai Barbuceanu, Chris Beck and especially, Micheal Gruninger, with whom I could always discuss ideas for the formalization of axioms in first order logic.

Thanks to the many members (past and present) of CMTE - Paul Seethaler, Ani (Ghazarian) Rock, David Reese, Paul Simak, John Ruggieri, Mike Serbinis and the many other grad students with whom I have had the pleasure of many discussions together - you always helped in making my long hours seem easier by making me feel part of the CMTE team.

Thanks to my parents for teaching me to strive from a very early age by instilling good study habits in their children. They insisted we be seen doing our "school work" daily at the dining table. To our pride and joy, Jason and Karen, thanks for attending to the many small, but important family life tasks at home and in the caring of their grandparents when Dad was away. Last, but not the least, I express sincere appreciation and gratitude to my dearest wife, Shuyee, for supporting me in this endeavour - it would not have been possible otherwise.

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 Introduction

1.1 Executive Summary

The Activity-Based Costing (ABC) Principle includes the assignment of costs to activities based on their use of resources, and the assignment of costs to "cost objects"[1] based on their use of activities [Brimson 91]. Since ABC assigns costs to activities based on their use of resources, the logical formulation of ABC must be premised upon the existence of some given or identifiable unit resource cost that must be associated with each resource required by an activity. Notwithstanding this obvious rationale for unit resource cost being absolutely essential for ABC, there has been little or no development in the field of ABC to solve *two fundamental problems relevant to ABC:-* **(i) What unit resource costs are associated with a resource? (ii) How does one deduce unit resource cost(s) so that direct, indirect and overhead costs are accounted for within the costs of a resource?** The solving of these two fundamental problems relevant to ABC has been the motivating drive and central focus of this research.

1. Within the ABC literature, the term "cost objects" refers to the reasons for which activities are performed in enterprises.

The holistic **solution** provided consists of 3 interdependent parts:-

1. *A Cost Ontology for Enterprise Modelling* that promotes a precise and unambiguous understanding of temporal cost behaviour of resources and activities for a formalization of the ABC Principle. It is based upon the TOronto Virtual Enterprise (TOVE) Model that is founded upon the TOVE Core Ontology for activity, state, resource and time. This part identifies and associates costs for a resource and their effects on the cost of an activity requiring the resource.

2. A *Theory of Resource Cost Units* that provides a logical and formalized framework to deduce the basic unit costs of a resource identified in 1, so that conventional concepts of direct, indirect and overhead costs are accounted for in the unit costs of a resource.

3. A *Prototypical Demonstration of a Cost Advisor* that encapsulates axioms, principles and theories of the two previous parts in an IT environment referred to as the TOVE Testbed. It provides a visualization of the "activity-based" enterprise modelling infra-structure and a work-bench to deduce solutions to some cost related queries, referred to as informal competency questions, as posed by our corporate partners - deHavilland Inc. (Bombardier), Toronto; Broken Hill Proprietary (BHP), Melbourne, Australia.

The thesis of this dissertation, then, is: **Costs within an organization can be described by representing them in an enterprise model to provide a Formal Model and a Micro-theory of ABC; and costs of a resource can be objectively deduced to include direct, indirect, and overhead costs for a populated enterprise model by applying a Micro-theory of ABC and a Theory of Resource Cost Units.**

The term "populated enterprise model" refers to the population of activity instances for an enterprise. In particular, the thesis is focused upon representing, deducing and reasoning with the costs of activity instances to achieve an activity-based analysis of the costs in the production of a product or service. It *assumes* that a "complete activity model" is provided for an activity instance. Secondly, that given this assumption, i.e., the completeness of the activity model, the identification of cost drivers is no longer relevant to the analysis. A forensic or predictive activity-based analysis of costs in the production of a product or a service is dependent upon the "complete activity model" representing activity instances that have occurred in the past or that will occur in the future respectively. Thirdly, it is imperative to note that the forensic or predictive activity-based costs of a product will depend on the time period of analysis. That is to say, for example, if the time period considered for the analysis is one year, as is normally done, the cost of a product based on the annualized analysis would be different from the costs of the product based on a quarterly or monthly basis, and vice versa.

The thesis is supported by the development of the following models:

- a formal, re-usable, descriptive model of enterprise costs, called a Cost Ontology for Enterprise Modelling in TOVE,
- a formal, re-usable, prescriptive model called a Micro-theory for ABC,
- a formal, re-usable, descriptive model called a Formalization of Overheads,
- a formal, re-usable, prescriptive model called a Micro-theory of Resource Cost Units.

1.2 A Need for this Research

According to the ABC concept, costing of "cost objects" proceeds with the assignment of cost to activities based on their use of resources, and the assignment of costs to "cost objects" based on their use of activities. Within the ABC literature, the term *"cost objects"* refers to the reasons for which activities are performed in enterprises. Products, services, and customers are considered cost objects as there may be reasons why activities are performed. This concept towards costing has gained wide and favourable acceptance in all sectors of industry in the US, UK and Canada [CMA 93].

First, the question is: From where and how does one get the costs of resources for ABC? Second, ABC emphasizes the need to obtain a better understanding of cost behaviour and thus ascertain what causes the overhead costs [Drury 89]. However, towards solving the question and fulfilling this need, there are some problems that influence the feasibility of the ABC Principle being applied to enterprises.

Let us examine the current costing process in ABC. According to Turney [92], ABC is accomplished in a two stage process. In the first stage, the cost assignment of resources to an activity is accomplished through a resource cost assignment phase through "resource drivers". In the second stage, the cost assignment of activities to cost objects is accomplished through "activity drivers". Resource drivers are the links between resources and activities. They take cost from the general ledger and assign it to activities. An activity driver is a factor used to assign cost from an activity to a cost object. According to Turney [92(A)]:

- "Various factors, referred to as resource drivers, are used to assign cost to activities."
- "Activity drivers are methods for assigning the cost of activities to cost objects. .... Cost objects are costed accurately when activity drivers measure the use of activities directly or correlate closely with their use. .... The objective is to pick the right number and type of activity drivers. Enough of the right type are needed to report accurate costs - though too many may be costly and may create a system that's too complex to understand."

Based upon the selection of *resource drivers* and *activity drivers*, "overhead and indirect costs" get assigned and included into the cost of a cost object through cost pools [Turney 92]. In other words, traceability and cost assignment of "overheads and indirect costs" to cost objects is dependent on resource drivers and activity drivers. Presently, this is the state of the art in ABC with regard to "overheads and indirect costs".

Therefore, given that there can be several different resource drivers and activity drivers, the cost of the activity is only as good as "a resource driver", and the cost of the cost object is only as good as "an activity driver".

In other words, the operationalization of the process, and the computation of the cost of an activity and the cost of a cost object are in themselves open to inconsistencies and inaccuracies. Does this reduce ABC to being an art, rather than a science? Maybe it does, as is implied in the following excerpts from [Babad & Balachandran 93]:-

> "An ABC system achieves improved accuracy in estimation of costs by using multiple *cost drivers* to trace the cost of activities to the products associated with the resources consumed by those activities. In this respect, a cost driver is an *event*, associated with an activity, that results in the consumption of the firm's resources."

> " 'The art of designing an ABC system can be viewed as making two separate but interrelated decisions about the *number of cost drivers needed* and *which cost driver to use*. These decisions are interrelated because the type of cost drivers selected changes the number of *drivers* required to achieve a desired level of accuracy' (Cooper 1989a, 1989b)."

In short, there is a need to operationalize the resource cost assignment and the activity cost assignment with better consistency, better accuracy, better traceability of "overheads and indirect costs", and less ambiguity. With regard to ambiguity, notice the confusing usage of the terms - *resource driver, activity driver, cost driver, event* and *drivers.*

This research demonstrates that the Formalization of ABC and the Theory of Resource Cost Units founded upon the Cost Ontology for TOVE is a major step towards fulfilling this need in the ABC field. First, the approach adopted towards formalization consists of Enterprise Modelling and Ontological Engineering. This approach enables us to communicate the ABC Principle as a formal model, where formal refers to a data model expressed using a logic language, not an analytical model expressed in mathematics. Unlike most English sentences, and like an equation written in mathematics, a logic language expression can be interpreted unambiguously, as long as the terms that comprise the expression are precisely defined. Second, the solution provides a clear mapping of resource drivers to resource cost units (i.e., cost of a unit of resource depending upon its state of existence in the world) for the resource cost assignment; and the mapping of an activity instance to an activity driver for the activity cost assignment. Third, the solution provides a Formalization of Overheads so that traceability, accountability, and assignability of "fixed overheads" (e.g., depreciation, property taxes, etc.) and "variable overheads" (e.g., cost of utilities- hydro, heat, water) is achieved with better consistency, better accuracy, and less ambiguity. Fourth, the solution provides a logical framework that encapsulates a Micro-theory of ABC and a Micro-theory of Resource Cost Units so that an objective deduction of resource cost units of a resource can be made to include direct, indirect and overhead costs directly to activities and to cost objects. Fifth, the cost of specifying an activity for a cost object is given consideration to be included in the cost of an activity called for by a cost object. It is noteworthy that cost of specifying an activity for a cost object is given little or no consideration in ABC developments to date. And lastly, based upon the logical framework presented, it is possible to automate the deductive reasoning of the resource cost units of significant resources of significant activities in a populated enterprise model.

1.3 The Costing Problem

The costing problem arises in that traditional cost systems themselves have major shortcomings. This problem is exacerbated in that existing enterprise modelling frameworks provide a modelling infrastructure that tend to support traditional cost systems [Tham & Fox 98]. Traditional cost systems use volume-driven allocation bases such as direct labour hours, direct machine hours, direct labour dollars, direct material dollars, and sales dollars as the primary means of assigning organizational expenses and overheads to individual products, services and customers. However, many of the resource demands by individual products and customers are not proportional to the volume of units produced or sold. Thus, traditional cost systems do not measure accurately the costs of resources used to design, to produce, to sell and to deliver products to customers.

The traditional cost systems may have proved adequate for providing product costs when the overhead costs and costs of indirect activities was a small percentage relative to the direct labour consumed by the products. However, modern enterprises are mostly operating in a *capital intensive and automated environment* with direct labour as a percentage of total operating cost being typically around 8 percent [Zhuang & Burns, 91]. A survey of electronic firms indicate a cost profile breakdown as being 13 percent for materials, 8 percent for direct labour, and 79 percent for overhead [Hawks et al., 1992]. Significant overhead costs such as equipment depreciation, cost of monies borrowed, property taxes, employee supplementary benefits, and insurance costs are apportioned amongst product lines using volume related rates such as direct labour hours. Therefore, by using direct labour as a primary apportioning device to distribute significantly high overhead and indirect costs to products, can cause significant distortions in product costs and poor strategic decisions.

In general, *the apportionment of indirect and overhead costs to products and service products based on volume related units such as direct labour or machine hours according to traditional cost systems provide irrelevant costs for decision making and for the determination of product or service profitability* [Cooper 89] [Kaplan 88].

Kaplan and Cooper of Harvard Business School have developed an approach to product or service costing called activity-based costing (ABC) as a means to overcoming some of the problems with traditional costing systems.

Traditional costing methods were designed primarily to satisfy the need for inventory valuation as defined by GAAP (Generally Accepted Accounting Principles) in manufacturing organizations. The objectives of traditional or conventional cost accounting systems are to provide historical cost information to serve two primary masters: first, government regulatory agencies, especially the tax bureaus such as Revenue Canada and the Internal Revenue Service of the USA; and second, the owners of the corporations, namely, the stockholders. Tax and other regulations such as GAAP specify very rigid rules for handling cost data so that quarterly and annual reports of corporations provide information in formats well understood by stockholders, financial analysts and potential buyers of a company. These systems are properly referred to as "financial accounting systems". Therefore, traditional costing systems narrowly focus on using cost information as a means to achieving financial accounting through the preparation of a variety of financial statements for external users.

We are faced with the issue that traditional cost systems use cost information to serve external users (i.e. non-company personnel); but they do not serve the needs of internal users (i.e., company personnel), especially the needs of managers and operations' personnel. According to Turney [92(A)], managers are reluctant to accept indirect costs from traditional cost systems as they provide "no information about activities". In order to make the assignment of indirect costs more acceptable to managers, Turney implies that the strategy should be for a cost system that provides relevant activity information that is traceable and timely.

The needs of managers and operations' personnel approximately fall into two areas in which cost information is used:-

1. The development of product costs that support pricing and strategic planning. This includes the identification of relevant costs as well as the distribution of indirect costs.
2. Management control systems that help in planning and reviewing. Included here is variance analysis arising from budgets and standard costs, as well as a variety of issues relating to measurement and evaluation of managerial performance.

The above two areas form the domain of "managerial accounting systems" in that the relevant rule or test becomes the usefulness of cost information for the management of the enterprise. Presently, the needs of managers and operations' personnel are not considered specifically in the traditional cost accounting systems. Cost information for their two areas of needs stated above must be dug out or "guesstimated" at best from the various monthly, quarterly or annual financial reports. This has left managers swamped with data while still starving for information that assists them in decision making.

The deficiencies in the traditional cost accounting systems for management decision making show up vividly and are most damaging in profit margin data for individual products or cost objects. Indirect or overhead costs are assigned arbitrarily to cost centres processing components of many varieties of products and involving mixtures of high and low cost activities that may require simple manual labour and expensive automation equipment. Profit margins using such costs allocated to individual products based on their direct labour content become not only absurd and meaningless, but are misleading when direct labour has diminished to less than 10 percent of total factory cost.

Traditional cost accounting systems have not given credence to the reality of high overhead or indirect costs as being non-volume related costs. In contemporary manufacturing enterprises, many significant indirect costs such as set-up, procurement, and quality control are all examples of costs where variations relate to such factors as diversity, complexity and degree of customizing of output. Therefore, to unitize costs such as these on a volume related basis, such as direct labour hours, introduces systematic bias into the resultant cost information; and, if the process is examined, it places the accountant in an indefensible position.

Two actions must be taken by enterprises facing these rising overhead costs: they must be attacked and reduced substantially in order for the company to remain competitive. To do so, new methods that promote the traceability and the assignment of overhead costs to individual products must be found and used.

1.4 ABC: Objectives and Issues

Traditional cost accounting allocates costs to products based on the attributes of a single unit of a cost object, a typical attribute being the number of direct labour hours expended to manufacture the unit. Allocations therefore vary directly with the number of units produced. In contrast, ABC focuses on the activities required to produce each product or to provide each service, based on activities consumed by the product or service.

Therefore, the ABC approach to costing is radically different from the traditional cost approach that tends to only measure direct labour and machine hours used for products; and then distributes a pool of indirect and overhead costs based on the direct labour and/or machine hours used for the products. The ABC concept of directing resource costs to activities (referred to as *resource cost assignment*), and the assignment of costs to cost objects based on their use of activities (referred to as *activity cost assignment*) is intuitively more appealing to one's common sense in terms of deriving costs of products effectively, efficiently and accurately [Turney 92(B)].

ABC has two primary objectives:-

1. to obtain more accurate data on product, process, and service costs;
2. to identify the costs of activities and the reason those activities are undertaken.

In both cases, the value of the information lies in its potential application by managers to help in everyday decision making. Decisions that are affected by cost information include pricing, new product introduction or abandonment, and the identification of productivity improvement opportunities.

The survey put forth by the Society of Management Accountants of Canada (CMA 93) indicates that the ABC concept has been widely and favourably accepted by all sectors of industries - manufacturing, retail, food, resource, commercial, energy, transport, and finance. It has been emerging as a widely accepted technique for general economic understanding of all overhead or indirect costs. However, notwithstanding the wide and favourable acceptance to the ABC concept over the last eight years, current ABC systems have had limited success in the US, UK and Canada. According to the survey [CMA 93] there have been very few implementations of ABC systems in these three countries - approximately 14 percent in Canada, 6 percent in the UK and 36 percent in the US based on a total sample size of 5,422 companies in the three countries over the period of 1991 to 1993 inclusive.

Even for those companies that have adopted ABC, there is the recognition that the potential scope of ABC extends beyond purely the production overhead category [Innes & Mitchell, 1991] in that ABC should include activity based information that *"comprises any relevant information about activities across the entire chain of value - design,* *engineering servicing, production, distribution, marketing and after sales service"* [Johnson 88]. Consequently, there is a need for new methods that provide information about the costs attributable to "indirect functions". [Borjesson 97].

According to Innis & Mitchell [91], within the ABC developments thus far, there is a *need* for the inclusion of 'overhead resource' and non-production overheads for activity based analysis. This is fully supported by others who emphasize the need for selling, distribution and service costs to be integrated within the ABC environment [Johnson & Kaplan 87] [Jeans & Morrow 89].

The lack of a "comprehensive cost unitization" of resources [Innes & Mitchell 91] *"across the entire chain of value"* makes it almost impossible for companies to achieve fully the decision oriented, control and 'scorekeeping' purposes for ABC based information towards the "balanced scorecard" [Kaplan & Norton 92].

Owing to lack of overhead cost traceability and hence its accountability, companies attempting to implement ABC form overhead cost pools for allocation to activities. Too often, different types of costs are combined into one diffuse overhead pool, so *product costs are often grossly distorted due to allocation* [Borjesson 97]. Tracing enables one to assign costs based on specific data, whereas allocation from pools often involves indirect assignment of costs to activities [Turney 92(A)].

The Emerging Practices in Cost Management [Brinker 96] indicates that, for ABC, if products are to be costed in a manner which reflects their actual consumption of resources, then there is a need for "overhead resources" to be formalized in a manner such that the costs of "overhead resources" can be traced to activities and allocated to products on the basis of time.

1.5 Enterprise Modelling Issues

According to Fox & Gruninger [97]:-

> "An Enterprise Model is a computational representation of the structure, activities, processes, information, resources, people, behaviour, goals and constraints of a business, government, or other enterprise. It can be both descriptive and definitional spanning what is and what should be. The role of an enterprise model is to achieve model-driven enterprise design, analysis and operation.
>
> From a design perspective, an enterprise model should provide the language used to explicitly define an enterprise. ....
>
> From an operations perspective, the enterprise model must be able to represent what is planned, what might happen, and what has happened. It must supply the information and knowledge necessary to support the operations of the enterprise, whether they be performed manually or by machine. It must be able to provide answers to questions commonly asked in the performance of tasks."

One way to effectively access and share information [Konsynski & McFarlan 90] towards costs of products, services, processes and systems of an enterprise is to represent and reason costs using information-systems based enterprise models [Kosanke 97].

To represent and reason about costs using an enterprise model, the model should be descriptive, i.e., it should represent key entities, structures and concepts needed to describe the enterprise's activities, resources, products, information flows and costs.

The model should also be prescriptive. It should be possible to prescribe the costs of activities, resources and products of an enterprise using this model. The ABC Principle is an appropriate cost prescription because:-

- it prescribes costs to products and services that give confidence to the enterprise regarding better accuracy and usefulness of information [CMG - IMA 94];
- it can address costs across activities and resources in all sectors of enterprises [O'Brien 90];
- it is one of the more important, widely accepted and publicized cost techniques towards Strategic Cost Management [Foster 96].

The models developed in this thesis are intended to be descriptive and prescriptive in that they represent knowledge data bases from which solutions to queries, referred to as competency questions, are prescribed through deductive reasoning.

A number of issues exist concerning the design of Enterprise Models [Fox 93]. The issues are:-

1. Reusability: it is concerned with the large cost of building enterprise-wide data models. Is there such a thing as a generic, reusable enterprise model whose use will significantly reduce the cost of information system building?
2. The Consistent Usage of the model: given the set of possible applications of the model, can the model's contents be precisely and rigorously defined so that its use is consistent across the enterprise?
3. Accessibility: given the need for people and other agents to access information relevant to their role, can the model be defined so that it supports query processing so that answers to common queries in an agent's domain (e.g., costing) may be obtained.
4. Selectivity: how does one know which is the right Enterprise Model for one's application?

Some of the major enterprise modelling frameworks are:-

1. Enterprise-Wide Data Model (EDM) [Scheer 89],
2. CIM-OSA (Computer-Integrated Manufacturing - Open-Systems Architecture) [ESPRIT 91],
3. Purdue Enterprise Reference Architecture (PERA) [Williams 92],
4. DoD (Department of Defense) Enterprise Model [DoD 93], and
5. CAM-I [Berliner et al. 88].

EDM, CIM-OSA, PERA and DoD tend to provide a modelling infrastructure that support traditional cost systems, whilst CAM-I provides principles for cost management in an advanced manufacturing technology (AMT) environment that sets the stage for ABC [Tham & Fox 98]

1.6 The TOVE (TOronto Virtual Enterprise) Project

At the Enterprise Integration Laboratory (EIL) of the University of Toronto, it is our belief that the issues of reusability, consistent usage, accessibility and selection can best be addressed by taking a more formal approach to enterprise modelling. *By formal, we are not referring to analytical models as found in Operations Research, but to logical models as found in Computer Science.* Towards this end, the TOVE (TOronto Virtual Enterprise) Project at EIL includes two major undertakings: the development of an Enterprise Ontology and a Testbed [TOVE92] [Fox et al. 93].

An ontology "consists of a representational vocabulary with precise definitions of the meanings of terms of this vocabulary plus a set of formal axioms that constrain the interpretation and well-formed use of these terms" [Campbell & Shapiro 95].

Ontologies are introduced in TOVE as a basis for modelling enterprises. The TOVE ontologies at the EIL currently span core knowledge about the enterprise such as activity [Gruninger & Pinto 95], time, causality, resource [Fadel et. al. 94], and organization [Fox et. al. 95]. These ontologies are collectively referred to as the TOVE Core Ontologies.

According to Fox [92], the objective of the TOVE (TOronto Virtual Enterprise) project is to create an enterprise ontology that has the following characteristics: (i) provide a shared terminology for the enterprise that every application can jointly understand and use; (ii) define the meaning (semantics) of each term in a precise and as unambiguous a manner as possible using First Order Logic; (iii) implements the semantics in a set of Prolog axioms that enable TOVE to automatically deduce the answer to many "common sense" questions about the enterprise; and (iv) define a symbology for depicting a term or the concept constructed thereof in a graphical context.

According to Fox et al.[93]:

> "We are taking what can be viewed as a "second generation knowledge engineering" approach to constructing our Common Sense Enterprise Model. Rather than extracting rules from experts, we are "engineering ontologies". The approach to engineering ontologies begins with defining an ontology's requirements; this is in the form of questions that an ontology must be able to answer. We call this the competency of the ontology. The second step is to define the terminology of the ontology - its objects, attributes, and relations. The third step is to specify the definitions and constraints on the terminology, where possible. The specifications are represented in First Order Logic and implemented in Prolog. Lastly, we test the competency of the ontology by "proving" the competency questions with the Prolog axioms. The TOVE ontologies constitute an integrated enterprise model, providing support for more powerful reasoning in problems that require the interaction of multiple ontologies."

The TOVE Enterprise Ontology provides a generic, re-usable ontology for modelling enterprises. The Core TOVE Enterprise Ontology spans knowledge of activity, state, resource, causality and time. The Core TOVE Ontology constitutes an integrated enterprise model, providing support for more diversified and powerful reasoning in problems that require the interaction of multiple ontologies that may span domain specific knowledge in the areas of organization, supply-chain management, quality and activity-based costing (as developed in this thesis report).

The TOVE Testbed provides an environment for analyzing enterprise ontologies. The Testbed provides a model of an enterprise (a lamp manufacturing plant), and a graphical user interface software tool called Oak [EIL94] developed in the Enterprise Integration Laboratory (IE Dept., U of T) for building, browsing, editing and visualizing a ROCK knowledge base [ROCK 92]. [ROCK: Representation of Corporate Knowledge; Carnegie Group's knowledge representation software tool developed under the IMKA - Initiative for Managing Knowledge Assets - project]. Oak also facilitates the deductive processing of queries input in Quintus Prolog [Quintus 91].

The data model for the lamp manufacturing plant pertaining to the core TOVE ontology is implemented on the TOVE Testbed on top of C++ using ROCK.

1.7 Proposed Contributions

The Cost Ontology for TOVE, the Micro-theory of ABC, and the Theory of Resource Cost Units directly contribute towards providing the solution to a fundamental question in ABC - From where and how does one get the costs of resources for ABC? In providing the answer to this fundamental question with a logical framework, this research provides the opportunity to automate the deductive reasoning of resource cost units to include direct costs, indirect costs, overhead costs, and the cost of specifying an activity for a cost object. This leads to better consistency, better accuracy, less ambiguity and promotes "open book accounting"[Foster 96].

The Cost Ontology for TOVE and Micro-theory of ABC provides a Formalization of ABC in terms of precise terminologies and axioms that enable us to deduce the temporal cost behaviour of resource, activity and cost object in a consistent manner compatible with the ABC Principle.

The Theory of Resource Cost Units enables us to deduce the costs of a resource. These costs are based upon the intrinsic state that the resource exists in. The theory enables us to deduce the resource costs with improved traceability and accountability of direct, indirect and overhead costs.

The Cost Ontology and Micro-theory of ABC serve as a foundation to establish an ABC Cost Advisor - a software that provides graphical and deductive solutions to cost related questions in a consistent manner compatible with the ABC Principle. The ABC cost solutions provided include direct, indirect and overhead costs that have direct traceability and accountability towards the costing of cost objects.

From the perspective of Enterprise Modelling, the Enveloped Activity Based Enterprise Model (EABEM) provides the infrastructure of the activities, states and resources of any enterprise based on the TOVE Core Ontology. The formulation of EABEM is made possible through the Principle of Resource Probing, the Principle of Activity Probing, and a Formalization of Overheads to represent, as "virtual" resources, entities associated with traditional notions of overhead. Unlike several other enterprise modelling efforts, EABEM provides the enterprise infra-structure that makes it possible to integrate and implement several of the Cost Management Principles of CAM-I for Activity-Based Management (ABM) and Strategic Cost Management.

From the perspective of successful ABC implementation, this research eliminates the need and confusion companies face in attempting to determine "the right cost drivers and the right number of cost drivers" to make the ABC system manageable and conducive for change management in an organization.

Finally, based upon the literature reviewed re: SAVILE (Ch. 2) from PricewaterhouseCoopers, the largest Accounting Firm according to Forbes Magazine, the formal models developed in this research may serve as some of the "rigid underlying models" that help generate the thousands of numbers typically summed up in their Financial Modelling Practices at that company.

1.8 Outline of Thesis

This chapter has provided an insight into the two fundamental problems with ABC **-(i) What unit resource costs are associated with a resource? (ii) How does one deduce unit resource cost(s) so that direct, indirect and overhead costs are accounted for within the costs of a resource?** It identifies the need for this research. It provides the 3 interdependent parts of a holistic solution. It has pointed the issues related to traditional costing and ABC. It briefly provides a description of the TOVE project and the enterprise modelling approach taken in the development of the solution. Finally, the proposed contributions of this research have been indicated.

Chapter 2 is titled Literature Review. It provides a review of enterprise modelling efforts undertaken over the past decade and the cost perspectives of these models. Cost management requirements are reviewed in the light of the Cost Management Principles of CAM-I that tend to support activity-based costing (ABC). Problems and issues relating to ABC are also reviewed. Other cost models such as Target Cost and Throughput Accounting are reviewed in the context of ABC. A more detailed overview of the TOVE Project is presented. The chapter also includes key points of four major ABM (Activity-Based Management) software packages, viz., Activa from PricewaterhouseCoopers, Profit Manager PLUS from KPMG Peat Marwick, EasyABC Plus from ABC Technologies, and NetProphet II from Sapling Corporation. In order to provide the reader some insight into "formal works" in the domain of cost accounting that may connect with contributions from this research, the chapter includes a review of the knowledge-based system SAVILE (PricewaterhouseCoopers), that is based upon a specific ontology appropriate for several reasoning tasks in accounting and auditing.

Chapter 3 puts forth the Methodology and Core Ontologies towards the development of the proposed solution presented in this research.

Chapter 4 details the design of the Cost Ontology for TOVE and a Micro-theory of ABC. It shows a clear mapping between the ABC concept and the Cost Ontology for TOVE.

Chapter 5 formalizes a Theory of Resource Cost Units. This is composed of a Principle of Resource Probing, a Principle of Activity Probing, the development of an Enveloped Activity Based Enterprise Model (EABEM), a Formalization of Overheads, and a Micro-theory of Resource Cost Units. It concludes with a logical framework for the objective deduction of resource costs in a populated enterprise model so that direct, indirect and overhead costs are consistently and accurately accounted for.

Chapters 6 and 7, demonstrates the implementation of the Cost Ontology and Micro-theory for ABC towards a Cost Advisor that is tested with case studies in deHavilland Inc. (Bombardier, Toronto, Canada) and Broken Hill Proprietary (BHP, Melbourne, Australia) respectively. The deHavilland case study is presented as a Forensic Cost Analysis with ABC in Ch.6; while the BHP case study of Ch. 7, titled Enterprise Modelling and ABC, demonstrates the manner in which the Cost Advisor provides forensic and "predictive what-if" solutions to several cost related competency questions.

In Chapter 8, the re-usability of the Cost Ontology is demonstrated in the light of the ABM functionality of the NetProphet II software package from the Sapling Corporation.

Chapter 9 provides the summary and contributions of this research, and outlines directions for future research that build upon the contributions of this thesis.

CHAPTER 2 Literature Review

2.1 Introduction

The focus of the thesis dissertation involves enterprise modelling and activity based costing. Its research directive is a Formalization of ABC in an Enterprise Model. The objective of this formalization is to lay the foundation for a holistic solution towards obtaining the costs of resources so essential for ABC. The term "holistic" implies that requirements and solutions are considered from a variety of viewpoints. To this end, the literature reviewed in this chapter gives coverage to the areas most pertinent for this holistic solution - enterprise models and their cost perspectives, cost management, ABC/ ABM, costing problems faced by enterprises, other cost models such as Target Costing and Throughput Accounting, an approach to enterprise modelling and integration, and ABC/ ABM software.

2.2 Enterprise Models & Their Cost Perspectives

Due to rapid changes of economic, geo-political and technological conditions, enterprises must be able to adapt to change with ease and quickness. This is called agility. To achieve an agile response to change, enterprise systems must be engineered, implemented and integrated in a systematic way very similar to approaches developed for software engineering. This has given rise to the emerging discipline - Enterprise Engineering [Vernadet 97]. According to Williams [97], Enterprise Integration is the coordination of the operation of all elements of the enterprise working together to achieve the optimal fulfillment of the mission of the enterprise as defined by enterprise management.

Enterprise modelling is a predominant activity in Enterprise Engineering and Integration. The 'Enterprise', in this context, is described as a socio-economic organization. It is comprised of a set of elements with a highly complex set of interactions. To be able to understand, improve and control the Enterprise, modelling techniques are required to deal with the complexities. Models provide a simplified representation or abstraction of reality. The model may include the representation of physical objects or concepts that help organize and unify knowledge. [Smart et al. 97]

According to Fox & Gruninger [97], an Enterprise Model is a computational representation of the structure, activities, processes, information, resources, people, behaviour, goals and constraints of a business, government, or other enterprise. It can be both descriptive and definitional spanning what is and what should be. The role of an enterprise model is to achieve model-driven enterprise design, analysis and operation.

Over the last decade, to improve enterprise integration, agility and competitiveness in the global markets, there have been several efforts in enterprise modelling to establish comprehensive representational frameworks of corporate knowledge with the expectation that a better understanding of shared corporate knowledge by diversified participants or agents of the enterprise would yield better costs and profits or "a better bottom line" as is commonly referred to.

Included in these enterprise modelling efforts are the more significant ones such as:-

1. Enterprise-Wide Data Model (EDM) [Scheer 89],

2. CIM-OSA (Computer-Integrated Manufacturing - Open-Systems Architecture) [ESPRIT 91],
3. Purdue Enterprise Reference Architecture (PERA) [Williams 92],
4. DoD (Department of Defense) Enterprise Model [DoD 93], and
5. CAM-I [Berliner et al. 88].

EDM, CIM-OSA, PERA and DoD provide insights into representational frameworks of corporate knowledge, whilst CAM-I provides principles for cost management in an advanced manufacturing technology (AMT) environment that sets the stage for ABC.

The modelling efforts - EDM, CIM-OSA, PERA and DoD - have concentrated on capturing the functionality or activity aspects of the enterprise being modelled without regard to addressing the problems in the management of costs for their products or services as put forth by CAM-I. They tend to support traditional cost accounting as opposed to strategic cost management which evaluates opportunities for repricing, the redirecting of resources to more profitable ends, and the re-engineering of processes in the production of products or the delivery of services as outlined by CAM-I. The methodology of the modelling efforts towards the representation of enterprise functionalities or activities have not provided the infrastructure to support strategic cost management promoted by CAM-I [Tham & Fox 98]

2.3 Scheer: Enterprise -Wide Data Modelling (EDM)

The enterprise-wide data modelling contribution by Scheer [85] was initiated in Germany in the mid 1980s and undertakes to construct data structures for typical functional areas (departments) such as production, engineering, purchasing, human resources, sales and marketing, accountancy and office administration that are generally encountered in an industrial enterprise with the aim of supporting planning, analysis, and traditional accounting systems in general.

Scheer uses the Entity-Relationship Model (ERM) [Chen 76] to systematically develop the *data structures* for the enterprise in terms of *entities* (something that can be identified in the users' work environment), *attributes* (characteristics/properties of an entity) and *relationships* (the association of entities with one another). The *temporal* (time) *aspects* [Allen 83] [Ladkin 86] of data structures are recognized in a rudimentary fashion through the identification of key attributes of entities as calendar time related, such as date, year, (fiscal) period and time-table-number. The temporally dependent processes, which are typically transaction data, are defined by their links with the special entity type TIME.

Knowledge representation is achieved through the identification of entities, their key attributes, and relationships with other entities from the areas of accountancy and/or from the other function areas. Entities, relationships and attributes are linked using Chen's ERM to construct a semantic network for data structures within accountancy. The abstraction mechanism of generalization, which enables an entity type to be thought of as a more generic entity type, is used to organize enterprise entities into a hierarchy through a partial order relationship, often referred to as the IS-A relationship [Brachman 85].

2.3.1 Cost Perspectives in EDM

Scheer's EDM view of *accountancy identifies five functional areas - Financial Accounting, Cost Accounting, Accounts Payable, Accounts Receivable, Payroll Accounting - that have connected entity relationships with one another and other functional areas of the enterprise.*

Scheer illustrates a sectorial integration principle [Scheer 85] whereby the sectors of purchasing, invoicing, and production data collection are related to Accounts Payable, Accounts Receivable, Fixed Asset Accounting, Payroll Accounting, and Inventory (Material) Accounting to achieve the General Accounting function for the EDM in the CIM environment [Scheer 91]. Scheer's EDM provides the infrastructure to fully support traditional cost accounting.

2.4 CIM - OSA: Enterprise Modelling

CIM-OSA is the acronym for Computer-Integrated Manufacturing - Open-Systems Architecture. The objective of Computer-Integrated Manufacturing (CIM) is the appropriate integration of enterprise operations by means of efficient information exchange within the enterprise with the help of information technology; and the Open-Systems Architecture (OSA) defines an integrated methodology to support all phases of a CIM system life-cycle from requirements specification, through system design, implementation, operation and maintenance, and even system migration towards CIM-OSA solution [Jorysz and Vernadat 90].

The European Strategic Programme for Research and Development in Information Technology (ESPRIT) supports a number of CIM related projects. CIM-OSA is the outcome of one such ESPRIT project, code number 688 with code name AMICE, the reverse acronym for European Computer Integrated Manufacturing Architecture.

2.4.1 CIM-OSA Enterprise Modelling Paradigm

CIM-OSA provides an architecture to describe the real world of the manufacturing enterprise by providing a unique set of advanced features to model functionality and behaviour of CIM systems at three distinct levels - *requirements definition, design specification and implementation description.* This description is used to control the enterprise operation and to plan, design and optimize updates of the real operation environment.

The CIM-OSA modelling methodology is more *descriptive than prescriptive*. In order to satisfy the needs of particular enterprises, CIM-OSA *does not provide* a standard architecture to be used by the whole manufacturing industry, but rather a Reference Architecture from which Particular Architectures may be derived. To facilitate this derivation of Particular Architecture from a Reference Architecture, CIM-OSA brings together a number of architectural principles and employs several structuring concepts [ESPRIT 91].

2.4.2 Cost Perspectives in CIM-OSA

CIM-OSA recognizes the importance of Enterprise Activities within the Business Processes of an enterprise. *CIM-OSA falls short* of adequately guiding the user, for example, in costing a product or supporting an investment decision based on a cost related criteria like profit. Perhaps the lack of a cost perspective of an enterprise may have been overcome if the developers of CIM-OSA included a view called the Cost View.

In summary, *much remains to be done or extended* in CIM-OSA in providing cost contents to the underlying concept of decision support across the centrally recognized construct Enterprise Activities within Business Processes. All CIM-OSA can do is to provide the Reference Architecture and its structuring means for organizing the different aspects of the enterprise. It is still up to the individual user to apply it in a *creative way* to improve the cost-effectiveness and efficiency of his/her enterprise and thereby make it more competitive in the marketplace.

2.5 The Purdue Enterprise Reference Architecture (PERA)

In December, 1990, PERA was particularly developed as an endeavour in enterprise modelling for a CIM factory by the Purdue Laboratory for Applied Industrial Control at Purdue University, West Lafayette. As early as mid 1986, the CIM Reference Model Committee of Purdue for Industrial Computer Systems had recognized *the need to establish a basis for the treatment of human-implemented functions* in a CIM enterprise. This need formed the focal point in initiating the PERA endeavour.

2.5.1 The Modelling Paradigm in PERA

PERA centres around representation of Information Systems Tasks, Manufacturing Tasks and the Human-based Tasks of the enterprise being modelled. The knowledge representation schema for each of these task modules is very similar to the Integrated DEFinition (IDEF) modelling approach [ICAM 81].] The generic representation schema incorporates Inputs with Time, Enabling Parameters, Transformation Process, Outputs and

Storage entities. For example, the enabling parameters for an Information Task are numerical parameters that enable algorithms to transform data input into computer outputs to be stored as databases.

2.5.2 Cost Perspectives in PERA

PERA provides no explicit representation of cost aspects in modelling an enterprise. Cost accounting aspects like standard product cost, actual product cost, expense variances, budget planning and profits *would* have to be modelled along the Information Stream as Information System Tasks or Processes using knowledge representation schema for Information Tasks. From a cost perspective, time stamped cost data would serve as inputs into the appropriate transformation cost algorithms based on numerical parameters to be considered for various computed outputs as per the cost accounting requirement. These computed cost outputs would serve as records within the stored database. This could provide the periodically updated cost related reports towards the cost control aspects of the enterprise mission.

2.6 The DoD Enterprise Model

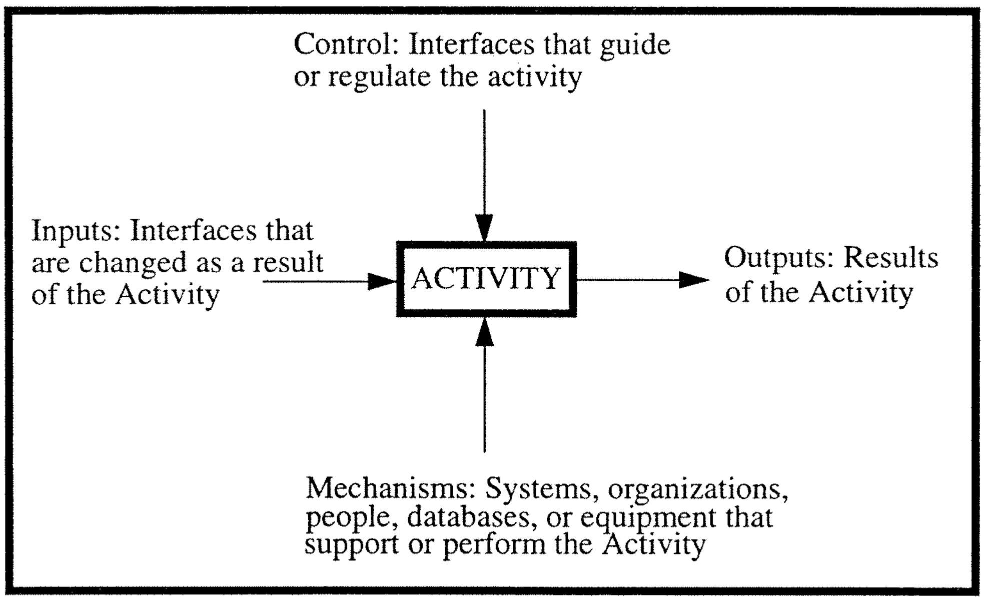
The DoD Enterprise Model [DoD 93] development is a currently on-going project undertaken by the U.S. Department of Defense (DoD) under the auspices of the President's National Security Strategy of the United States of America. With the end of the Cold War, the U.S. DoD has increased its emphasis on domestic issues to attain greater effectiveness and efficiency in all defense activities, with the hope that strategies developed in the process will positively impact the industrial competitiveness of the U.S. in the global markets of the 1990s and beyond. The DoD Enterprise Model, as we know it today, is the outcome of such an endeavour.

2.6.1 Model Concept and Overview

The DoD Enterprise Model serves as a guiding framework for all missions and functions undertaken by the DoD. Each activity of the DoD Enterprise Model is modelled using the Integrated DEFinition (ICAM 81) or IDEF modelling approach.

The DoD Data Model constitutes the data component of the overall DoD Enterprise Model which integrates process and data into one cohesive and coherent frame of reference. The Data Model was developed in conjunction with the IDEF Activity

FIGURE 1 An IDEF Activity Model



Model (Figure 1) to analyze inputs, controls, outputs and mechanisms of activities. The Data Model incorporates the Entity-Relationship (E-R) [Chen 76]] approach for knowledge representation purposes.

From the basic construct of an activity as shown in the figure, a model of an enterprise is developed and analyzed through a top-down process of decomposition and inter-relationships amongst activities as per the Structured Analysis and Design Technique (SADT) to construct the IDEFO activity model diagrams [Ross 77].

2.6.2 Cost Perspectives in the DoD Enterprise Model

The present definition of the DoD Enterprise Model mainly addresses the concerns of logically integrating activities and data, and linking them to DoD mission areas.

The bulk of cost accounting aspects related to the defense operations are concentrated in the Manage Acquisition activity. This activity performs the management and decision support functions needed to successfully implement approved acquisition programs and plans. It provides the program manager functions of fiscal baseline control, contracting and the fiscal administration of contracts. It monitors the delivery of goods and services, approves payments, prepares and transmits bills, and evaluates contract performance. However, it falls short of providing a comprehensive framework for the cost management aspects of the activities themselves. Some of these cost management aspects of activities are the tracing and computation of costs to activities and changes in costs if activities are re-engineered. If the DoD Enterprise Model is to serve as a modelling paradigm for US industries to improve their global competitiveness, *DoD must extend all efforts to provide a costing framework of generic principles, axioms, cost entities and their relationships so that all activities within the DoD Enterprise Model are cost managed and cost accountable.* This framework would then serve as an activity cost management model for the US industries in general.

2.7 CAM-I: Cost Management Systems (CMS)

2.7.1 CAM-I Background

In 1986, Computer Aided Manufacturing-International, Inc. (CAM-I) of Arlington, Texas formed a consortium of progressive private sector manufacturing enterprises, professional accounting firms, government agencies, universities and academia. Today CAM-I stands for Consortium for Advanced Manufacturing - International.

The main role of this consortium is to define the role of cost management in the advanced manufacturing technology (AMT) environment by providing an international forum where cost *management experts could share ideas, experiences and expertise to consolidate their knowledge about practices that have proven successful in the proliferating advanced technology environment.*[Berliner and Brimson, 88].

2.7.2 CAM-I Identifies a Need for CMS

A UK survey of management accounting in advanced manufacturing environments published by CAM-I in 1988 and a US survey sponsored by the National Association of Accountants and CAM-I reported that both preparers and users of cost accounting information were dissatisfied with their product costing systems. The survey concluded that the major area of concern was identified as the method of charging overheads to products. Advanced manufacturing technologies have revolutionized the manufacturing shop floor. Consequently, *there are dramatic changes in manufacturing cost-behaviour patterns: the direct labour and inventory components of product cost are decreasing, while depreciation, engineering and data-processing costs are increasing* [Berliner et al. 88]. These changes have resulted in higher overhead rates and a shrinking base of labour over which the allocation of overhead costs are made as is the case in traditional costing. Managers find it almost impossible to function as decision makers in an environment where cost accounting practices contribute to overhead rates being so high as to obscure true product cost, and where other accounting practices hinder manufacturing decisions. At the same time, the thrust towards "free trade agreements" has made international competition more vigorous. Managers are in need of up-to-date and concise information, formatted to assist them in making timely and correct decisions. CAM-I concluded that, in order to alleviate company struggles with important economic issues [Cunningham 92], there is an urgent need for Cost Management Systems - systems that must provide cost information necessary for informed operational and strategic decisions about resource acquisitions and their usage in an AMT environment. *The concept of CMS in enterprise modelling took form from CAM-I.*

2.7.3 Definition of CMS

CAM-I defines a cost management system as a management planning and control system with the following objectives:-

- to identify the cost of resources used in performing the significant activities of the enterprise;
- to determine the efficiency and effectiveness of the activities performed (performance measurement);
- to identify and evaluate new activities that can improve the future performance of the enterprise (investment management);
- to accomplish all of the above in a changing technology environment (manufacturing practices).

2.7.4 CMS Principles

The goal of CMS is *to provide information to help enterprises use resources profitably to produce services or products that are competitive in terms of cost, quality, functionality and timely introduction into the world markets.*

Several guiding principles for the conceptual design of cost management systems were identified by CAM-I. Even though these principles, in general, are compatible with the existing cost accounting framework as acknowledged by the National Association of Accountants (NAA) in their study entitled Management Accounting in the New Manufacturing Environment (NAA, 1988), taken as a total system, *these principles do represent a significant departure from the objectives and focus of existing cost accounting theory and practices in the following key concepts:-*

- While cost accounting takes a historical perspective and focuses on reporting costs, the main focus of *CMS is to take a pro-active role in planning, managing and reducing costs* with the reporting of actual cost results as part of a feedback loop only to modify a plan to improve execution as conditions will inevitably undergo some changes in the interim period between planning and execution.

- CMS should provide *the necessary and timely cost information for informed operational and strategic decisions about resource acquisition and use.*
- CMS should incorporate **activity based accounting,** *which is the collection of financial and operational performance information about significant activities of the business.*
- CMS should provide a framework for *continual improvement in the elimination of non-value added (NVA)* activities.
- CMS should *improve traceability of costs to management reporting objectives.*

2.7.5 Cost Management in the 1990s

In October, 1989 the Institute of International Research organized - and Coopers and Lybrand hosted - a conference in UK which sought to address the cost management issues. This Conference concluded that:

> "Whilst the subject matter in the conference was varied and covered a large number of business problems regularly confronting management, such as make-versus-buy decisions, product-profitability measurement and strategic investment appraisal, a common thread running through the conference was that of activity-based costing (ABC).
>
> This technique can no longer be considered an impracticable idea or the result of the fertile mind of theorists, but instead should be thought of as a very necessary tool to be used by an increasing number of businesses both large and small either to supplement current costing systems in a decision support role or to replace entirely conventional cost accounting systems."

According to Cokins [96], the cost management framework relevant for today is that which was created by Dr. Peter Turney and Norm Raffish from their involvement with the CAM-I Consortium. This framework is often referred to as the *CAM-I cross*. The vertical of this cross represents the cost assignment view and direction of ABC, i.e., costs of resources to activities via resource drivers for activity costs, and costs of activities to cost objects via activity drivers. The vertical of the CAM-I cross is also referred to as the *cost object view*. The horizontal of the cross provides the *process view* - ABM, to explain why things cost and what causes costs to exist. The cost object view intersects the process view at activities, activities being the operative word in ABC and ABM.

2.8 ABC: Activity-Based Costing

An accurate product cost is essential to make correct decisions in setting product prices and in selecting the optimal product mix. Global price competition [Porter 80] [Porter 85] has forced companies to question the accuracy of their product costs. At the same time, academicians such as Prof. Kaplan (Harvard Business School) published early warnings that multiple cost systems were necessary to obtain accurate product costs [Kaplan 88]. Today, ABC is the system discussed as the most effective method to obtain accurate product costs [Stoffel 92].

2.8.1 Background

In their 1987 book publication of *Relevance lost, the Rise and Fall of Management Accounting*, Johnson & Kaplan [87] argue that, although costing techniques developed in the 19th century were aimed at cost management, costing systems lost their relevance as cost accounting came to be dominated by the demands of financial accounting. External financial reporting required by the inclusion of overhead in stock and the primary criteria in the overhead allocation process became consistency, objectivity and economy rather than usefulness for decision making.

The provision of product cost information is one of the main purposes of a cost accounting system. Product costs play an important role in setting selling prices. With companies striving for global competitiveness [Porter 80] [Porter 85] [Booth 92], selling prices are determined by competition and are outside the control of the supplier. A firm that knows the costs of its products can concentrate on the most profitable product mix and avoid loss-making activities [Cunningham 92].

As a means to calculate product costs, Professors Robin Cooper and Robert Kaplan of Harvard Business School have developed a new approach called activity-based costing [Cooper 88a][Cooper 88b]. *The* **main premise** *of ABC is resource consuming activities cause costs and products incur costs through activities they require for design, engineering, manufacturing, distribution, etc.* An ABC system focuses on where costs come from without regard to allocation. ABC directs management's attention to underlying causes (drivers) of cost and product.

There have been critics of the fundamental assumption on which ABC is based, i.e. that activity causes cost. More notably, Piper & Walley [90] have argued that the **proposition** "*that decisions cause cost is superior to the argument that activity causes cost as decision precede activity*".

Let us discuss the **main premise** of ABC and the **proposition** put forth by Walley and Piper to gain some insights towards a solution to the Costing Problem of Ch.1. First, the operative word in activity-based costing is the word activity. In the ABC literature, it is unclear as to what represents an activity. By this we mean, what are the entities, characteristics, attributes and constraints that constitute an activity in all situations. This issue about "activity" itself is further troublesome when one examines the following excerpted definitions that bring about different aspects or dimensions to an activity:-

1. "Activities - Those actions required to achieve the goals and objectives of the function": CAM-I Glossary [Berliner & Brimson 89]. This definition has a goal driven dimension. Questions that arise from this:- Does this imply that actions that do not achieve goals and objectives are not activities? How did such activities come to be? Did they come to be because of some "decision" that preceded that activity? The preceding "decison" - is it merely to do or not do the activity? Or, how it should be done or not done? Then, it is perhaps correct to agree with Piper and Walley, that decisions cause cost!

2. "Activity - A unit of work performed within an organization. A description of the work that goes on in the organization and consumes resources. Testing material is an example of an activity." Glossary of Turney [92]. This definition is perhaps less

restrictive than the previous one. However, it introduces the aspect of resource consumption. Questions that arise from this:- What is it that constitutes our understanding of something being a resource to an activity?

3. "Driver - An activity or condition that has a direct influence on the operational performance and cost structure, or both, of other activities": CAM-I Glossary [Berliner & Brimson 89]. This definition brings the dimension of one activity influencing another activity in some way. Questions that arise from this:- Is there some entity of one activity that influences another activity?

In short, there is need to define, describe and represent an activity and a resource with more precision, consistency and less ambiguity.

2.8.2 Several Costing Problems Identified

The costing problems faced by companies towards their products, their services and in the generation of cost information for strategic decision making arise on several fronts.

I. Traditional or conventional cost systems adopted by the vast majority of enterprises, (almost 80 percent according to one survey [Nicholls 92]), *do not measure accurately* the costs of resources used to design, to produce, to sell and to deliver products to customers [Cooper & Kaplan 92]. This inaccuracy in traditional cost systems is caused by usage of volume-driven allocation bases such as direct material and direct labour hours as the primary means of apportioning high overhead costs to individual products, services and customers [Turney 92]. According to Gary Cokins [96]: "Accountants count what is easily counted, not what counts. Outdated, traditional accounting blocks managers and employees from seeing the relevant costs."

However, in many modern manufacturing enterprises, overheads are not homogeneous in terms of being primarily influenced by volume. The view shared by many is that most important contemporary overheads are largely unaffected by alterations to production volume. The overheads falling into this category mainly represent the costs of service support functions which are prevalent in enterprises to assist the efficient production of a range of quality products [Miller & Vollman 85]. In general, *the apportionment of indirect and overhead costs to products and service products based on volume related units such as direct labour or machine hours according to traditional cost systems provide irrelevant costs for decision making and for the determination of product or service profitability* [Cooper 89] [Kaplan 88]. Even though ABC addresses this problem to some extent, for purposes of the feasibility of the ABC concept being applied to enterprises to cost the "cost objects", there is a *need* to obtain a better understanding of *cost behaviour* and ascertaining the *causes of overhead costs* [Drury 89].

II. Nothwithstanding the fact that activity-based costing (aka ABC in short) does overcome some of the shortcomings of traditional cost systems through its treatment of non-volume related overhead costs, ABC developments have been primarily *restricted* to production overheads such as procurement, set-up, engineering services, maintenance and quality control in the manufacturing sector [Innes & Mitchell 91].

III. Even for those companies that have adopted ABC, there is the recognition that the potential scope of ABC extends beyond purely the production overhead category [Innes & Mitchell, 1991] in that ABC should include activity based information that *"comprises any relevant information about activities across the entire chain of value - design, engineering servicing, production, distribution, marketing and after sales service"* [Johnson 88]. Consequently, there is a need for new methods that provide information about the costs attributable to "indirect functions". [Borjesson 97].

IV. *"Activity"* being the operative word in ABC, the costing problems are exacerbated as some of the major enterprise modelling frameworks that tend to represent activity information of enterprises *"across the entire chain of value"* regrettably support traditional cost systems and *do not provide* an adequate activity infra-structure that promotes ABC through CAM-I's Principles of Cost Management Systems (aka CMS Principles) [Berliner & Brimson 89] [Tham & Fox 98]. Included in these modelling efforts are Scheer's Enterprise-Wide Data Model (EDM) [Scheer 85], CIM-OSA (Computer-Integrated Manufacturing - Open-Systems Architecture) [ESPRIT 91], Purdue Enterprise Reference

Architecture (PERA) [Purdue 92], DoD (Department of Defense) Enterprise Model [DoD 93], and CAM-I (Consortium for Advanced Manufacturing-International).

V. Within the ABC developments thus far, there is a *need* for the inclusion of 'overhead resource' (term coined by Innes & Mitchell) and non-production overheads for activity based analysis [Innes & Mitchell, 1991]. This is fully supported by others who emphasize the need for selling, distribution and service costs to be integrated within the ABC environment [Johnson & Kaplan 87][Jeans & Morrow 89].

VI. The lack of a "comprehensive cost unitization" of resources [Innes & Mitchell 91] *"across the entire chain of value"* makes it almost impossible for companies to achieve fully the decision oriented, control and 'scorekeeping' purposes for ABC based information towards the "balanced scorecard" [Kaplan & Norton 92].

VII. Even for companies that have embraced ABC, the accurate analysis of profitability by market segment, product line, distribution channel and customer which underlies business strategy formulation is dependent on ABC being extended to include some measure of cost apportionment of overheads such as rent, interest rates, insurance, building depreciation, power, heat and light [Innes & Mitchell 91].

VIII. Notwithstanding that the ABC concept has been well defined, has wide and favourable acceptance in all sectors of industries in the US, UK and Canada [SMA 93], the literature and developments towards ABC system implementation thus far seem to involve terminologies and procedures which confuse and make the task of ABC implementation difficult for the user. To illustrate this point, the following cited items 1, 2 and 3 are brought to the attention of the reader:-

1. The ABC Principle includes the assignment of cost to activities based on their use of resources through resource drivers, and the assignment of costs to *"cost objects"* based on their use of activities through activity drivers [Brimson 91][Turney 92]. According to Burk and Webster, a *resource driver* is a measure of the consumption of a resource, used to determine the portion of the total resource cost assigned to each activity that uses the resource; an *activity driver* is a measure of the frequency

of activity performance and the effort required to achieve the end result; and a *cost driver* is an indicator of why an activity is performed and what causes the cost of performing the activity to change [Burk & Webster, 1994].

2. " .... ABC is based on a two stage procedure. First, charging overhead cost to activity based cost pools. Second, deriving and using a series of cost driver based rates to attach the pooled costs to product lines. Its design and operation is therefore dependent upon three key factors: the choice of cost pools; the selection of means of distributing overhead cost to the cost pools; and the choice of cost driver for each cost pool. These factors represent the basic mechanics of an ABC system......." [Innes & Mitchell, 1991].

3. "In principle, a two-stage approach is used in ABC product costing. The thrust of the design, as codified originally by Robin Cooper, is to identify a relatively small set of both volume-sensitive and nonvolume-sensitive overhead cost drivers (say six to twelve) and to trace indirect costs to each driver. Then the company determines the percentage of the drivers consumed by each product or service. The result is an estimate of the indirect costs of each product based on nonvolume-sensitive drivers such as engineering change notices (ECNs), setups, and inspections, as well as the traditional volume-sensitive drivers such as direct labor hours and material dollars.

   The 'drivers' referred to here resemble both the 'activity drivers' and the 'activities' found in GE's early cross-functional activity cost analysis. In the ABC literature, the word "activity" often is used synonymously with "driver," although activity is the word that has stuck to describe the nonvolume-sensitive driver-based product costing technique." [Johnson 92].

Cited item 1 implies that the ABC principle includes a two stage procedure of costing, viz., (i) an activity is costed based on resources required, and (ii) a cost object is costed based on activities called for; cited items 2 and 3 above put forth the two stage basic mechanics of an ABC system as bearing little or no resemblance in terminologies and procedures as those of item 1. Also the terms and procedures indicated in item 2 differ from those of item 3. In summary, cited items 1, 2, and 3 illustrate some evidence of a lack of formalization in terminologies for the communication of relevant cost data and a lack of a "formalized theory" that ensures the fundamental computations for the consistent application of *resource cost assignment* and the *activity cost assignment* within the ABC paradigm such that one may deduce the costs of the cost objects.

IX. There is the aspect of overhead cost control - through allocation or elimination? Automation and an increase in production volume and support functions are often said to have contributed to the explosive growth of overhead cost. Are these the "real" causes of overhead? Not necessarily. Many overhead costs are caused by activities and transactions resulting from managerial maladjustments and breakdowns in the normal functioning of the organization. Cost allocation of overhead costs should be considered only when elimination, reduction, or traceability is not possible, and the allocation should be in congruence with the company's goals and objectives. *When elimination of overhead cost is not feasible, the next best alternative is to improve its traceability and hence the accountability* [Tatikonda & Tatikonda 91]. The Emerging Practices in Cost Management seems to suggest there is *a need for the improvement of overhead cost traceability and hence its accountability* [Brinker 96]. *Cost should be traced wherever possible and allocated only as the last resort* [Turney 92(A)]. New methods of overhead cost control are needed. [Borjesson 97].

X. Owing to lack of overhead cost traceability and hence its accountability, companies attempting to implement ABC form overhead cost pools for allocation to activities. Too often, different types of costs are combined into one diffuse overhead pool, so *product costs are often grossly distorted due to allocation* [Borjesson 97]. Tracing enables one to assign costs based on specific data, whereas allocation from pools often involves indirect assignment of costs to activities [Turney 92(A)].

XI. The Emerging Practices in Cost Management [Brinker 96] indicates that in order to achieve strategic company goals, overhead costs should be allocated on the basis of lead time as was suggested by [Schonberger 86]. This may sound radical to most traditional accountants, but in reality this is a fair procedure as in many cases overhead cost is a result of time and effort expended in dealing with delays and problems caused by increased lead times in delivering products and services to customers. Therefore, for ABC, if products are to be costed in a manner which reflects their actual consumption of resources, then there is need for "overhead resources" to be formalized in a manner such that the costs of "overhead resources" can be traced to activities and accounted to products on the basis of time.

XII. Due to global competition, companies are striving to be the lowest cost producer of their products and services. Given the increase of indirect costs as a proportion of total costs, *indirect costs offer the potential for large cost reductions*. However, companies have to overcome the overhead dilemma - the mismatch between the urgent need to *reduce* indirect costs and the *lack* of appropriate cost information [Blaxill & Hout 91] [Borjesson 97]. New ideas and methods for cost reduction as part of profit control is seen as essential to the strengthening of international competitiveness [Tai 92].

2.8.3 Factors Leading to the Development of ABC

The main factors [Jeans, 1989] which led to the development of ABC are:-

- management accounting practice has become distorted by the needs of financial reporting; in particular, costing systems are driven by the need to value stock, rather than to provide meaningful product costs;
- direct labour has shrunk to less than 10 percent of total factory cost [Plossl 92] for the majority of manufacturing companies, yet direct labour is extensively used as a common basis of loading overheads onto products;
- overhead costs, conversely, have increased greatly and fixed components predominate;
- the marketplace being more competitive, every business has a need to understand product costs and know what drives overhead through a cost management system that supports process improvement and performance measures aligned with strategic objectives.

2.8.4 Issues about ABC Implementation

ABC surveys conducted in Canada [SMA 93], USA [Cunningham 92] and UK [Nicholls, 92] indicated that over 50 percent of respondents were unclear of the best way forward towards ABC implementation. According to Nicholls, "The ABC approach, although stated as 'simple' by many authorities on the subject, is clearly not considered to be simple to implement."

The UK survey [Nicholls, 1992] also summarized some of the *expected* significant benefits from ABC implementation as being:-

- justification (i.e., composition and traceability) of overhead charges of products and services supplied to customers requiring 'open book' costing,
- ability dynamically to model our businesses with effects on costs.

In case study discussions with Robin Cooper and Robert Kaplan, Mike Jeans and Michael Morrow of Peat Marwick McLintock's Cost Management Group point to the following issues [Morrow 89] that will arise when introducing ABC principles into a costing system:-

1. Where should the boundary fall for the inclusion of overhead costs?
2. What cost drivers should be used?
3. How can all costs be attributed to products at the unit level?
4. What computer systems are there to run ABC on?

2.9 Other Cost Models

Besides ABC, there are at least two other cost models being deployed by companies for purposes of achieving strategic cost management. The first is Target Costing (TC), and it is widely applied in Japanese industries. The second is Throughput Accounting (TA), and it has helped corporations such as Garret Automotive of UK to double its profits from the same sales volume without any reduction in its personnel head count [Darlington et al. 92].

2.9.1 Target Costing (TC)

Cost management in Japan is guided by Target Costing [Berliner & Brimson 89]. Unlike North America, cost management is the responsibility of engineers, not accountants. The Japanese treat costs as a symptom, not a cause or a solution. However, Japanese industries embrace cost symptoms as important clues for seeking opportunities or tackling problems [CAM-I: TC 97].

TC begins with the assumption of the customer's ability to pay for a product [Sakurai 89]. That is, TC starts with market-based pricing independent of cost. And since earning a profit is considered as a given for businesses, then a target cost simply becomes a calculated number that cannot be exceeded during a product's design and start-up phase. A Target Cost for a product represents a market-based cost that is calculated using a sales price necessary to capture a predetermined market share and a desired level of profit [Berliner & Brimson 89].

Target Cost = Sales Price (for the market share) - Desired Profit

If the target cost is initially lower than the budgeted or standard costs, cost reductions would be factored into the budget and standards over a period of time. Cost reduction can be achieved in two ways: (i) a learning curve occurs during start up production volumes as a process is being fine tuned; (ii) the company applies a philosophy of continuous improvement (aka Kaizen in Japanese) in eliminating waste.

According to Monden & Hamada [91]: "the management accounting system in Japan is functioning very well through Target Costing and Kaizen Costing for Japanese automobile companies. .... (i) 'Target costing' is the system to support the cost reduction process in the developing and designing phase of an entirely new model, a full model change or a minor model change. (Target costing is called 'Genkakikaku' in Japanese.) (ii) 'Kaizen costing' is the system to support the cost reduction process in the manufacturing phase of the existing model of product. (Kaizen costing is called 'Genkakaizen' in Japanese.)"

Target costing and Kaizen costing are closely coupled for cost management in Japan. To implement the Target Costing concept, a company may want to develop more detailed measurement systems for activity-level costs and performance. Such systems would help identify progress in meeting the overall target cost objectives as the company finishes with Kaizen during the subsequent production phase.

2.9.2 Throughput Accounting (TA)

While ABC is a technique which focuses primarily on overhead costs, TA provides an approach which takes overhead and labour costs as given and concentrates on the flow of production through the factory. *Throughput* is defined as the rate at which raw materials are turned into sales. That is, *throughput is defined as sales less material costs per period of time.*

The throughput approach was publicized by Goldratt & Cox [84] in their book, The Goal - Excellence in Manufacturing, which was written in the style of a novel. TA is closely coupled with The Theory of Constraints(TOC) by Goldratt [90]. In many manufacturing and process businesses, there will be one part of the process that may constrain the capacity of the whole enterprise. Capacity on the bottleneck operation becomes very valuable if demand is in fact straining the capacity of that operation. Measuring product profitability must recognize the extent to which a product uses the bottleneck operations. Until additional capacity is obtained, any new product will have to incur an opportunity cost of the contribution lost by the displaced product. Thus, traditional costing systems may well be inaccurate and misleading where there are constrained bottlenecks. This new view of costs brings greater emphasis to material flow and has spawned the name *throughput accounting.*

Essentially, TA recognizes that capacity constraints are gating factors to making profit. That is to say, any time lost at a bottleneck is forever lost to the total business and results in lost profit. Conversely, any time gained from removing a *nonbottleneck* is only a mirage with no bottom-line impact despite the extra effort.

Some of the indices [Waldron & Galloway 88] used in TA are:-

Return per factory hour = (Sales Price - Material Cost) / Time on key resource

Cost per factory hour = Total factory cost / Total time available on key resource

TA ratio = Return per factory hour / Cost per factory hour

TA or TOC advocates criticize ABC because the numbers from ABC vastly differ from that of TA. Since TA supports just-in-time (JIT) and total quality management (TQM) philosophies that go with JIT, to TA advocates, ABC numbers appear both wrong and bad, and do not support the Lean Enterprise [Cooper 96].

The discrepancy in TA and ABC numbers may be explained. The TA or TOC advocates assume that much or all of the overhead cost allocations of a plant can be loaded at the bottleneck or constrained work centre. This assumption leads to the escalation of the cost of any product that uses the constrained work centre, which conversely reduces costs to similar products going through *nonbottlenecked* work centres. Consequently, TA calculations yield vastly different product costs relative to ABC. TA clearly penalizes products "renting time" at the bottleneck [Cokins 96]. Cokins [96] suggests that the TA cost measures be used to understand the direction where incremental product profit may come from and to aid future planning for capital or resource spending.

2.10 An Approach to Enterprise Modelling and Integration

This section gives further insight into the "philosophy" and directives at EIL for the TOVE Project. This has been a major influencing factor into the methodologies (Ch. 3) followed in this research.

Over the last 10 years our perceptions of the operations of an enterprise have evolved from one of being "hierarchical in structure and control" [Fox 81] to one of being distributed over inter-dependent organizational agents (humans or machines) that communicate and cooperate in problem solving and action [Fox et al. 93].

According to Fox et al.[93]: "Enterprise integration is concerned with the means to improve the performance of disparate agents of distributed organizations and markets. The focus of enterprise integration is on the communication of information, the coordination and optimization of enterprise decisions and processes for improved productivity, competitiveness, flexibility and quality. Integration necessarily warrants that agents of the enterprise be able to understand one another. One of the principal paths towards achieving integration is the use of information technologies (IT) to provide access to information, to support decision making and to assist in action execution. It is this path which has come to be known as the Computer Integrated Enterprise (CIE) and is one of the main focuses at the Enterprise Integration Laboratory (EIL) at the University of Toronto, Industrial Engineering Department."

Fox et al.[93] emphasize that: "To achieve enterprise integration, human or machine based agents of the enterprise must be able to understand each other. Therefore, there is a definite need for a language in which enterprise knowledge can be expressed. Minimally the language provides a means of communicating among agents in sales, marketing, manufacturing, design, etc. Maximally the language provides a means for storing knowledge and employing it within the enterprise for purposes of production control, quality control, cost management, etc."

2.10.1 Language and Representation

According to Fox et al.[93], there is a distinction between a language and knowledge representation. A language is commonly used to refer to means of communication among people in the enterprise. Representation refers to the means of storing information (aka knowledge) in a computer (e.g., database). A representation is essentially a set of syntactic and semantic conventions that enables one to form a knowledge repository or database in a computer for usage by various agents in a distributed systems environment. The set of syntactic conventions specify the form of the notation used to express descriptions of the knowledge entities. The set of semantic conventions specify how expressions in the notation correspond to the entities described. With the advent of computer based distributed systems, there is need for various processes (aka agents) to communicate directly with one another. Consequently, the representation of knowledge has become the language of communication.

According to Fox et al.[93], due to this consequence, companies face three problems today. Technological advancements and competition today have led to independently created computer systems to support enterprise functions. This makes it almost impossibe to share the same representations. This has led to different representations of the same enterprise knowledge, and only further promotes the inability of enterprise functions to share common enterprise knowledge. Secondly, these representations are defined without an adequate specification of what the terminology means (aka semantics). This leads to inconsistent interpretations and uses of the enterprise knowledge. Lastly, current representations are passive in so far as being unable to automatically deduce the obvious from what it is representing. An example cited by Fox et al.[93] is a representation that contains a 'work-for' relation, where it is explicitly represented that Joe 'works-for' Fred, and that Fred 'works-for' John. Then the obvious deduction that Joe 'works-for' John (indirectly) cannot be made within the representation system. The lack of ***"common-sense"*** deductive capability forces users to spend significant resources on programming each new report or function that is required.

2.11 Measurement Criteria for Knowledge Representation

The success of an existing representation of knowledge can be measured in two ways: 1) the extent of the intrinsic characteristics of *generality, efficiency, perspicuity, transformability, extensibility, granularity, scalability, minimality* and *competence* [Fox & Tenanbaum 90] [Fox et al. 93] of the representation, and 2) the extent of *re-usability* of the representation.

2.11.1 Intrinsic Characteristics of a Representation

The first measure of success of a representation focuses on the following intrinsic characteristics of the representation [Fox et al. 95] [Gruber 93]:

- Generality: To what degree is the representation shared between diverse activities such as design and troubleshooting, or even design and marketing?
- Efficiency: Space and inference. Does the representation support efficient reasoning, or does it require some type of transformation?
- Perspicuity: Is the representation easily understood by the users? Does the representation "document itself?"
- Transformability: Can the representation be easily transformed into another more appropriate one for a particular decision problem?
- Extensibility: Is there a core set of ontological primitives that can partitioned; or do they overlap in denotation? Can the representation be extended to encompass new concepts?
- Granularity: Does the representation support reasoning at various levels of abstraction and detail?
- Scalability: Does the representation scale to support large applications?
- Minimality: A minimal set of terms should be in the ontology.
- **Competence:** *How well does it support problem solving? That is, what questions (aka competency questions) can the representation answer or what tasks can it support?*

2.11.2 Re-usability and Reducibility

The second measure of success of a representation is the extent to which the existing representation (aka native ontology) is re-usable. One may view *re-usability* from two perspectives: 1) the *applications' perspective*, and 2) the ontological perspective based on the *reducibility concept.*

The *applications' perspective* of re-usability is based on the extent to which the native ontology can successfully perform tasks in the knowledge domain for two or more enterprises.

According to Gruninger[96], the *reducibility concept* is the extent to which other representations (aka target ontologies) of the knowledge domain can be expressed in terms of the native ontology to do some of the tasks performed by the target ontologies. Reducibility essentially involves mapping the semantics between target and native ontologies based upon their intended interpretations of the terminologies and axioms of the ontologies. Assume that a target ontology can perform some tasks in some enterprise, say X. Then, based on the reducibility concept, if the target ontology can be reduced to the native ontology that can do similar tasks, that would be evidence that the native ontology is a usable candidate for X for those tasks, thereby promoting its re-usability.

2.11.3 Competency of a Representation

According to Fox [92], the competence of a representation defines the types of tasks that the representation can be used in. The obvious way to demonstrate competence is to define a set of questions that can be answered by the ontology. If no inference capability is to be assumed, then question answering is strictly reducible to "looking up" an answer that is represented explicitly. In defining a shared representation, a key question then becomes: should we be restricted to just a terminology? Should the terminology assume an inheritance mechanism? Artificial Intelligence knowledge representations and object-oriented representations assume at least inheritance as a deduction mechanism. Or should we assume that some type of theorem proving capability is provided, say, in a logic programming language with axioms restricted to Horne clauses (i.e., Prolog)? What is the *deductive capability* that is to be assumed by an ontology? Fox[92] proposed that for each category of knowledge, a set of questions be defined that the ontology can answer. Given a representation and an accompanying theorem prover (i.e., Prolog), questions can be posed in the form of queries to be answered by the theorem prover. Given that a theorem prover is the deduction mechanism used to answer questions, the *efficiency* of a representation can be defined by the number of LIPS (Logical Inferences Per Second) required to answer a query.

2.12 ABM Software

According to Borden [94], "(ABM) Software is nothing more than a tool to help a company with its cost management efforts. By itself, no cost management software can solve an organization's cost management problems." There is a dearth of ABM software packages on the market. To limit the scope, ABM software reviewed fall into two distinct categories:- (i) software developed and marketed by one of the "Big" accounting firms like PricewaterhouseCoopers and KPMG Peat Marwick to support their consulting engagements; (ii) software packages designed to be installed and implemented by the purchaser. Software packages included in the first category are Activa from PricewaterhouseCoopers, and Profit Manager Plus from KPMG Peat Marwick. Software packages included in the second category are EasyABC Plus from ABC Technologies, and NetProphet II from Sapling Corporation. The rest of this section summarizes key points of each of the software packages reviewed.

2.12.1 Activa from PricewaterhouseCoopers

Key points [Albright 95] are:-

- Activa is built on the UNIX operating system and Oracle 7 database, thus combining the flexibility of PC-based tools with the computing power scaleable to mainframe environments,
- Activa uses Window Graphics User Interface,
- Activa supports traditional cost accounting and ABC accounting techniques,
- Activa incorporates multiple currencies to support international implementations,
- Activa allows interfacing with a user's current systems such as general ledger, operational, and order management systems.

2.12.2 Profit Manager PLUS from KPMG Peat Marwick

Key points [Borden 94] [Albright 95] are:-

- It was designed by Robin Cooper and Robert Kaplan.
- It is offered as a DOS, Windows and a network version.
- The design architecture uses terms and theory commonly found in ABC literature.
- Resources are associated with General Ledger Accounts, and are driven to activities with resource drivers.
- Activities are classified as direct or indirect activities. Indirect activities are driven to other indirect or direct activities by the use of resource drivers.
- Location drivers are used to pull activities into user-defined centres; whereas activity drivers pull resources out of centres to cost objects (products or components).
- It can generate custom reports, and exports to other software, including spreadsheets.
- It can perform capacity analysis, create activity-based budgets, and perform what-if analysis.

2.12.3 EasyABC Plus from ABC Technologies

Key points [Turney 92] [Borden 94] [Connolly & Ashworth 94] are:-

- Its design features are heavily influenced by the work of CAM-I and Peter B.B. Turney of Cost Technology (Oregon, USA).
- Implementing EasyABC involves developing a model using the CAM-I approach to generate a bill of activities, or an output measures approach which uses the bill of activities as an input. Each model is hierarchical in nature and is made up of components called centres and blocks. The nesting of centres and blocks is referred to as decomposition.

- A cost element is the lowest-level cost maintained by a block. Costs are distributed from one block to one or more other blocks through allocation. Allocation is done using either specific percentages or specific cost drivers as is suggested by Turney [92].
- It includes an Overhead Module, an Activity Module, and a Cost Object Module. The Overhead Module contains overhead expenses (e.g., salary, benefits, taxes) that are to be allocated to activities based upon some specific cost drivers or based upon some percentages. In the Activity Module, the user organizes the activities that are consumed by products. In the Cost Object Module, users define the cost objects to which they want to trace costs. In short, the modules are made up of components called centres and accounts, activities, or cost objects. The three modules constitute an ABC model for EasyABC.
- After an ABC model is built, cost data and driver data for a time period needs to be entered. Cost allocations are performed between modules based upon this data of costs and drivers.
- It provides a graphic user interface; and provides reports such as Bill of Activities, Bill of Costs, and Performance Measures for each centre.

2.12.4 NetProphet II from Sapling Corporation

This software is dealt with more extensively for purposes of demonstrating Re-Usability (Ch. 8) through Reducibility of the NetProphet "pseudo-ontology" to that of the Cost Ontology in TOVE. Key points [Albright 95] are:

- NetProphet does not have an "ontology" as we understand it at EIL.
- It is a Windows program containing a logical arrangements of icons, which helps in constructing efficient models.
- An industrial engineering perspective pervades in its design as there is a strong operational influence that considers both financial and operational data. This permits ABC, capacity planning, business process engineering, and strategic planning.

- What-if scenarios can be analyzed in terms of costs, revenues, and capacity.
- NetProphet permits data transfers from mainframe sources, and also permits transfers into other application software such as Excel and Lotus 1-2-3.

The review of ABM software packages confirms that the approach to ABM in this research is unique in that the Ontological Engineering approach to Enterprise Modelling for the Cost Domain facilitates ABM through the deductive reasoning of solutions sought for by companies to their "common sense" or informal cost queries (competency questions) for purposes of strategic cost management.

2.13 SAVILE: Ontology for Accounting & Auditing

According to Hamscher [94] of PricewaterhouseCoopers (PwC), the company wishes to develop knowledge-based system for their auditors involved in issuing opinions on the fairness of the financial statements of their clients. Being the largest accounting firm in the world, the company has identified the need to develop formal knowledge-based systems to support their auditors in recognition that:

> "Auditing tasks are difficult in part because auditors must understand the relationship between a perturbation in an accounting system and its effects, which can be highly indirect because accounting systems are complex. Fortunately, accounting systems are composed of many interacting processing elements that from an audit standpoint are conceptually simple, and this decomposability suggests that a model-based approach is appropriate: the user would build a computational model of the client system to automate the analysis of the effect of local perturbations."

Although a model-based approach has been proposed for financial tasks before, the models have generally focused upon relationships amongst variables representing financial and micro-economic quantities [Bouwman 83] [Hart et al. 86] [Bailey et al. 90] [Bridgeland 90] [Hamscher 91].

SAVILE by PwC is the first program that demonstrates a new model-based approach to supporting auditing tasks based upon an ontology and reusable models. According to Hamscher [94]:

> "SAVILE repesents the steps involved in producing and processing financial data in a company, using an ontology appropriate for several reasoning tasks in accounting and auditing."

There are three key points behind the ontology and its realization in the SAVILE implementation: to support simulation, to exploit existing ontologies, and to transform models to perform multiple tasks. This ontology is the foundation of the new model-based, formal and reusable model approach taken by PwC for SAVILE in the company's on going effort to develop knowledge-based systems to support their auditors. SAVILE is an implemented program that demonstrates the adequacy and appropriateness of this ontology of financial data processing for the evaluation of internal controls, the designing of tests, and the planning of other related audit tasks performed by auditors. The ontology has been founded upon the following three sources:-

1. The primary source was an experimental program called TICOM designed to support a certain class of queries about actions [Bailey et al. 85] [Bailey et al. 89].
2. A secondary source was guidance materials called AGS written within PWC for the use of audit staff.
3. An ontology called SEADOC used by a different accounting firm and has important similarities [Elliott 83].

The SAVILE ontology is implemented in a language SPLAT: SAVILE Programming Language Attributed to TICOM. Syntactically, this is a set of class definitions in a frame language JOSIE [Nado et al. 91]. The semantics of activities and the actions of which they are composed are an elaboration of Petri nets, supporting an interpreter that does event-driven simulation. Example of a SPLAT Action terminology for the ontology is: "**get** *record repository*", which means: "Extract a record from the repository".

According to PwC, the SAVILE ontology has now evolved into a larger and more extensive ontology encapsulated into a proprietary PwC "Accounting Tool" called COMET. PwC is presently negotiating patents for COMET and, therefore, cannot provide any specific literature on COMET.

In conclusion, SAVILE demonstrates a practical approach to model formalization and reusability grounded upon ontology development in the domain of accounting and auditing. Similarly, the Formalization of ABC in this research has been based upon the Cost Ontology for Enterprise Modelling (Ch. 4). According to Laurie Woodruff, Head of Financial Modelling, PwC, Canada, although financial statements of their clients consist of numbers that are typically summations of "thousands of other numbers", PwC is always looking for and dependent upon "rigid underlying models" that generate those "thousands of other numbers" based upon their clients' enterprise operations. The formalized models developed in this research may serve as some of the "rigid underlying models" sought after by PwC.

2.14 Directions for Enterprise Modelling and Cost Management

At EIL, our review of Enterprise Modelling and Cost Management concluded [Tham 93] [Tham & Fox 98] that, from a common sense standpoint, the organizational and functional composition of any enterprise is the integration or sum total of:-

1. disparate agents (human or machines) that must necessarily communicate with one another, that is, an agent must be capable of receiving and using comprehensible information from other agents, and must also be capable of transmitting comprehensible information for usage to other agents;
2. domain specific defined activities deemed necessary to be performed through various "states of completion" for planned and unplanned reasons to accomplish enterprise objectives;

3. the fiscal management and control features that guarantees the existence of the enterprise itself must be logically in unison with the above two ingredients.

The enterprise modelling projects - EDM [Scheer 85], CIM-OSA [ESPRIT 91], PERA [Purdue 92], DoD [DoD 93] - reviewed in this chapter do not go far enough in explicitly recognizing and incorporating the above. Hence, integration of the enterprise, for which the enterprise modelling efforts were primarily expended for, have fallen short of expectations.

If enterprise integration is to be significantly enhanced, research and development in enterprise modelling and cost management using ABC must necessarily be *co-developed for convergence* and must pave in-roads in the following areas:-

1. The creation of a generic and shared representation of corporate knowledge within each agent domain and/or otherwise, so that each disparate agent may communicate and use with one another in the distributed enterprise. By generic and shared representation we mean that the representation can be used for different domains and applications; and the representation can be easily understood by different agents of an enterprise.

2. The development of a precise, consistent and complete activity and state (stage of completion of an activity) centered representation of appropriate enterprise activities deemed necessary for execution towards enterprise mission fulfillment

3. The development of cost modelling framework closely linked with the enterprise activity knowledge representation to facilitate the dissemination of relevant cost information to various agents within a company.

One of the contributions of this research is the bridging of the gap between the domain of *enterprise modelling* and the *domain of cost management using ABC (aka ABM for Activity-Based Management).* The methodologies (Ch. 3) deployed and the developed solution presented in this research is focused upon bridging the gap with co-developed technologies related to AI (artificial intelligence), "which concerns the development of theories and techniques required for a computational engine to efficiently perceive, think, and act with intelligence in complex environments" [Fox 90].

2.15 Concluding Remarks

The enterprise modelling projects - EDM [Scheer 85], CIM-OSA [ESPRIT 91], PERA [Purdue 92], DoD [DoD 93] - have concentrated on capturing the functionality or activity aspects of the enterprise being modelled without regard to addressing the overhead allocation problem and principles in the management of costs for their products or services as brought forth by CAM-I.

CAM-I has identified several Principles for Cost Management Systems to assist in improved cost management. CAM-I pointed out that *activity based accounting* should be employed to improve the traceability of activity data needed for investment justification and benefit tracking to the specific investment opportunity.

The main premise of ABC is resource consuming activities cause costs and products incur costs through activities performed for them. ABC has been widely accepted as the *necessary tool* for effective cost management. However, the ABC approach, although stated as 'simple' by many authorities on the subject, is clearly not considered to be simple to implement.

The expectations of companies from ABC implementations are:-

1. justification (i.e.,composition and traceability) of overheads to their products and services supplied to customers demanding "open book" costing;
2. their ability to dynamically *model their businesses with effects on costs.*

These expectations call for the convergence or integration between the domain of enterprise modelling and the domain of cost management using ABC.

CHAPTER 3 Methodology & Core Ontologies

3.1 Introduction

This chapter presents the methodologies and core ontologies used to developing the 3 inter-dependent parts of the solution outlined in Ch. 1. The purpose of this chapter is to tutor the reader into understanding the semantics of an object oriented model of the activity cluster with its resources. It is the semantics of this model in TOVE that facilitates a formalization of the well accepted ABC Principle.

First, the terms ontology, micro-theory and advisor are described.

Second, the Ontological Engineering Methodology deployed for designing the Cost Ontology and Micro-theory of ABC is discussed. The design factors for an ontology are discussed with emphasis on the criteria of generality and competence of an ontology. Competence of an ontology is explained to be evaluated through "competency questions". The concepts of Re-usability and Reducibility are discussed for the evaluation of satisfactory generality of the ontology.

Third, the Core Ontologies of activity, state, resource, time, and causality are described as they are the building blocks for the Cost Ontology. The constraints on the semantics of the Core Ontologies are explained with a Foundational Theory grounded upon a dynamically changing world of situations through the *situation calculus*. A table of core terms, i.e., predicates of the Core Ontology for activity, state, and resource is presented together with their relationships.

3.2 Ontology, Micro-theory, and Advisor

Since the solution presented in this research relies heavily on the design, analysis and construction of prototypical implementations of the Cost Ontology for Enterprise Modelling in TOVE and a Micro-theory for ABC, the terms, ontology, micro-theory, and advisor are described with definitions and characterizations.

3.2.1 Ontology

Definition: An ontology is a formal description of entities and their properties; it forms a shared terminology for the objects of interest in the domain, along with definitions for the meaning of each of the terms [Gruber 93]

According to Gruber [93], the ontological engineering process involves the following:-

1. associating human-readable terms in the universe of discourse (the domain) to computer-readable *objects or classes*, *relations* and *functions*; and
2. stating formal *axioms* that constrain the interpretation and proper use of the terms in the domain.

Objects or classes refer to a collection of entities categorized as one because they share common properties; *relations* refer to relationships between these entities; and *axioms* give precise expression to the basic logical properties of entities in the ontology.

In practical terms, the ontological engineering process is described as creating a data dictionary of terms of a domain, simply referred to as terminology, and stating axioms that define and constrain the terminology.

One of the first steps in creating a data dictionary is the development of a classification hierarchy of terms referred to as a *taxonomy.* A taxonomy is a tree structure displaying **is-a** relationships [Brachman 82] in a hierarchical manner. Properties are inherited between the entities linked to by is-a relationships. A taxonomy may be seen as an important tool for ontological engineering as it helps organize the terms within the domain, and associates implicit information about the terms.

In regard to the stating of axioms, the *concept of minimal ontological commitment* is useful in practice. This concept is a way of bounding the number of axioms to be restricted to a minimum number of axioms required to minimally describe the domain [Newell 82] [Gruber 93]. This concept is useful in practice as, notwithstanding some axioms are shown, it is conceivable that many more axioms can be stated.

Ontology-based models are useful because:-

> "All knowledge-based systems refer to entities in the world, but in order to capture the breadth of human knowledge, we need a well-designed global ontology that specifies at a very high level what kind of things exist and what their general properties are." ([Rich & Knight 91], pg. 292)

3.2.2 Micro-theory

Definition: A micro-theory is a formal model of knowledge required to solve a problem in a domain or describe a subset of the domain in detail. It is separate from, but constructed upon, an ontology [Lenat & Guha 90].

As a micro-theory is based upon an ontology, it too contains terms and axioms. Therefore, it must be ensured that the way the terms are used and the meanings of the terms in these axioms are consistent with the ontology of the domain. The concept of minimal ontological commitment is applicable to a micro-theory.

3.2.3 Advisor

Definition: An advisor is a software tool which encapsulates, and enables performing tasks using ontologies and micro-theories [Fox & Gruninger 94].

The tasks for an advisor can be classified in *evaluation*, *analysis*, and *guidance*. *Evaluation* requires the ability to compare two different enterprise models along a dimension, such as cost or quality, and to evaluate that one model is better as per that dimension. *Analysis* tasks require prediction, monitoring, identification, and explanation. For example, in the cost domain, the analysis tasks by a cost advisor may involve predicting the cost of an activity, monitoring its cost with the passage of time, identifying cost components that compose the cost of the activity, and explaining the causes for the cost of the activity. *Guidance* implies that the advisor is capable of suggesting alternatives. For example, can an enterprise be guided by a cost advisor regarding the different sets of activities that may bring a project cost on budget?

An advisor is a useful tool in the development of enterprise designs. For example, using an advisor, a decision-maker would have the capability to pose queries to the populated enterprise model, and to manipulate the enterprise model to test out what-if scenarios. This allows for quick prototyping and analysis of alternate enterprise designs.

3.3 Ontological Engineering Methodology

The goal of enterprise engineering is the implementation of an environment that supports the modelling and design of enterprises. In order to support this, ontological engineering deals with the design and evaluation of a shareable representation of knowledge that minimizes ambiguity and maximizes understanding and precision in communication in an enterprise. The product of ontological engineering is an ontology, which is a formal description of objects, properties of objects, and relations among objects.

An ontological engineering methodology deals with the design of an ontology or micro-theory. Since a design must satisfy some factors or criteria, design factors are stated at the outset. For example, Gruber [93] states clarity and coherence as being two of many design factors that should be satisfied by an ontology. These factors can be designed into an ontology or micro-theory by following a rigorous, and iterative process, so as to engineer, rather than craft, an ontology or micro-theory.

The ontological engineering methodology put forth by Gruninger & Fox [95a] for EIL provides a template for a rigorous and iterative process so as to engineer an ontology to satisfy design factors. Fox et. al [93a] state generality and competence as being two of the design factors to be satisfied by an ontology. These criteria are used as part of the ontological engineering methodology in testing a designed ontology. How does one test an ontology for the criteria of generality and competence? Evaluation of generality may be operationalized by demonstrations of re-usability. Competence evaluation may be achieved through demonstrations of competency. Re-usability is discussed later in this section.

The ontological engineering methodology adopted at EIL provides a framework that guides the design of ontologies and the evaluation of competency of these ontologies [Gruninger & Fox 95]. The steps in the methodology are as follows:-

1. Interactions are sought and promoted with industrial partners like deHavilland and BHP to become familiar with problems that arise in their particular enterprises. This generates the motivating scenarios for the applications.

2. These problems define the ontology's requirements in the form of queries that an ontology must be able to answer. We term these queries as competency questions.

3. The ontology's terminology, that is, its objects, attributes and relations, are then defined. This provides the language to be used to express the definitions in the terminology and the constraints required by the applications.

4. Definitions and constraints on the terminology are specified wherever possible. These specifications are represented in First Order Logic and implemented in Prolog.

5. The competency of the ontology is tested with respect to its adequacy in solving the competency questions.

6. Extensions, improvements or proposals for new ontologies are sought through iterations of the above steps.

In addition, such a framework allows a more precise evaluation of different proposal for an ontology, by demonstrating the competency of each proposal with respect to the set of competency questions generated from the applications. The six step framework was the method applied to designing the Cost Ontology for Enterprise Modelling (Ch.4). It begins with queries in the cost domain that result in a Cost Ontology and Micro-theory for ABC. The success of providing solutions to the queries serves as an evaluation of the competency criteria being satisfied by the ontology.

3.3.1 Re-usability

Re-usability is the capability to use portions of a model to solve different problems beyond the problems that initially motivated the development of the model.

Evaluation of re-usability on an objective basis is also an issue. Objectivity is possible in regard to software engineering if some quantitative metrics are established. For example, according to Cusamano [91], Toshiba uses the ratio: percentage of re-used lines of code to actual lines that are included into a program with little or no modification. Another measure considered an extension of the Toshiba measure involves the re-use of entire software objects rather than re-use lines of code [Banker et al. 93]. Such quantitative empirical evidence is one of the several methods suggested by Guida & Mauri [93] for the evaluation of knowledge-based systems. Other methods of evaluation of re-usability are based upon empirical evidence, ground in knowledge-based systems life-cycle, original evaluation criteria, imported evaluation criteria, automatic knowledge-based checking methods, and effort estimation using analogy [Shepperd & Schofield 96].

Another model evaluation method is **Reducibility** as proposed by Gruninger [96] of EIL. The *reducibility concept* is the extent to which other representations (aka target ontologies) of the knowledge domain can be expressed in terms of the native ontology to do some of the tasks performed by the target ontologies. Assume that a target ontology can perform some tasks. Then, based on the reducibility concept, if the target ontology can be reduced to the native ontology that can do those tasks, and keeping in mind that the native ontology design was unaware of those tasks of the target ontology, then this would prove that the native ontology can be "re-used" for the "unknown tasks" of the target ontology. *In short, reducibility is the mapping of the semantics between target and native ontologies subject to the intended interpretations of the terminologies and axioms of the respective ontologies.*

The concept of Reducibility is used as a means to demonstrating the Re-usability of the Cost Ontology designed in this research.

This research considers the application of the reducibility concept as a means to prove the re-usability of the Cost Ontology, i.e. the native ontology, with that of an "implicit target ontology" for the software Netprophet II (Sapling Corporation) that offers an approach for developing an ABC model [Albright 95].

3.4 Core Ontologies

The effort in creating an enterprise model is fraught with problems. One is the Correspondence Problem [Fox 92] that must deal with the relationships among concepts that denote the same thing but have different terminological descriptions. It is common for enterprises, especially those that cross geographical boundaries to use different names to refer to the same concept. No matter how rational the idea of renaming is, organizational barriers impede it. Another problem is the sheer size of the model caused by models being created based on descriptive forms of relationships and concepts [Fox 92]. The descriptive forms themselves may promote ambiguity, which in turn promotes verbosity for clarification purposes of the descriptive forms, which then leads to increase in model size. If one were to develop a complete model for an enterprise based on the paradigm of descriptive forms of relationships and concepts such as in the ICAM (Integrated

Computer-Aided Manufacturing) model [Martin et al. 83] [Martin & Smith 83], its sheer size would be beyond the abilities of any database manager or knowledge engineer to understand and use effectively. Consequently, the creation of an enterprise model for a particular enterprise may have to be performed in another way.

Fox [92] recommends that the creation of a firm's enterprise model be a by-product of the enterprise design function. At EIL, our view of the enterprise model is similar to that of the IDEF (Integrated DEFinition) family of modelling languages in that it is the design of the firm's activities that entail a subset of enterprise modelling classes for instantiating, that is, a process of linking a real thing in the world to a concept [Sathi et al 85]. However, to successfully generate an enterprise model, the activity modelling methods must be more explicit in the specification of goals, activities, constraints, resources, states, etc. than is currently found in IDEF-like modelling paradigms. This has been the basis of formalizing ontologies or representations for the **core entities of time, activity, state, and resource** that form the architectural elements of creating an enterprise model in TOVE.

The Core Ontology in TOVE include the core entities of time, activity, state, resource, causality and organization structure. The Core Ontology contains the building blocks for the Cost Ontology, the Micro-theory of ABC, and the Micro-theory of Resource Cost Units presented in this report.

3.4.1 Foundation Theory of the Core Ontologies.

According to Gruninger[97] of EIL:-

> "The focus of the ontology is not only on the terms themselves, but also on their definitions. We can include an infinite set of terms in our ontology, but they can only be shared if we agree on their definitions. It is the definitions which are being shared, not simply the terms themselves.
>
> The problem is that the meaning of the terminology for many ontologies are in people's heads; we need some framework for making it explicit. Any ideas that are implicit are a possible source of ambiguity and confusion. For a process ontology,

the model theory provides a rigorous mathematical characterization of the process information and the axioms give precise expression to the basic logical properties of that information in the language.

It is useful to distinguish two types of sentences in this set of axioms: foundational theories and definitions. A foundational theory is a set of distinguished predicates and functions together with some axiomatization. Distinguished predicates are those for which there are no definitions; the intended interpretations of these predicates is defined using the axioms in the foundational theories. All terminology in a generic ontology is defined using classes of sentences in the foundational theories; any terminology that does not have a definition is axiomatized in some foundational theory. So when we speak about a semantics for an ontology, we are referring to the axiomatization of foundational theories and definitions for the ontology's terminology."

Within TOVE, the foundational theory of the *situation calculus* provide the semantics to the ontology of time, activity, and state, that are founded upon the First-Order Language (FOL) for representing dynamically changing worlds, called the situation calculus [McCarthy & Hayes 69].

The situation calculus is a sorted language based upon the sorts - actions, situations, time-points, and fluents [Gruninger 97]. According to Gruninger [97]:

"The intuition behind the situation calculus is that there is an initial situation (denoted by the constant S0), and that the world changes from one situation to another when actions are performed. The structure of situations is that of a tree; two different sequences of actions lead to different situations. Thus, each branch that starts in the initial situation can be understood as a hypothetical future. The tree structure of the situation calculus shows all possible ways in which the events in the world can unfold. Therefore, any arbitrary sequence of actions identifies a branch in the tree of situations. .... the function **do(a,s)** is the name of the situation that results from performing action a in situation s. There is a predicate **poss(a,s)** that is true whenever an action a can be performed in a situation s. ....

A **fluent** is a relation or function whose value may change between situations. To define the evaluation of the truth value of a sentence in a situation, the predicate **holds(f,s)** is used to represent the fact that some fluent f is true in situation s. ....

The foundational theory for TOVE also includes the extension of the situation calculus ..... in which one branch of the *situation tree* is selected to describe the evolution of the world as it actually unfolds and time points are associated with the start and end of each situation in a branch. The predicate **actual** specifies those situations which are in the actual branch, and the predicate **occurs(a,s)** is defined to represent actions performed along the actual branch."

In situation calculus, each perturbation to the modelled world changes the world from situation to situation. In this model are terms that describe an entity or relationships between entities of a domain in the world.

Fluent: If the truth value of a term in the ontology varies from situation to situation, then the term is a fluent. We say a fluent *holds* in a given situation if the term is true in that given situation. All terms defined in the Cost Ontology are fluents. A fluent occurs in a given time period, if for all the situations which occur during that period, the fluent holds.

Sit Calc Term: Pred-1. **s**

s: the world is thought of as being in some situation s; this situation can change only as a result of some agent performing an action.

Sit Calc Term: Pred-2. **f**

f : fluents are relations whose truth value may vary from situation to situation

Sit Calc Term: Pred-3. **holds (f, s)**

f : a fluent: a predicate whose truth value may change

s: change only as a result of some agent performing an action.

Sit Calc Term: Pred-4. **holdsT(f, t)**

f : a fluent: a predicate whose truth value holds true at time t

Sit Calc Term: Pred-5. **occursT(f, t)**

f : a fluent: a predicate whose truth value may change

t: time at which f holds true at time t

actual(do(a,s)): By definition, this predicate implies that the situations *s* do arise in the real world, and it is always possible to do the action *a* in each of these situations. This definition may be stated in FOL as follows:-

$$\textbf{\textit{Axiom:}}\ (\forall a, s),\ actual(do(a,s)) \supset actual(s) \wedge poss(a, s) \qquad \text{(FOL 1)}$$

occurs(a, s): By definition, this predicate represents action occurrences of those actions that are performed along the actual branch of a situation tree.

$$\textbf{\textit{Axiom:}}\ (\forall a, s),\ occurs(a, s) \equiv actual(do(a,s)) \qquad \text{(FOL 2)}$$

do([a1, a2, a3, ..., a(n-1), an], S0): represents a *legal action sequence* **[Gruninger 95]** where *a1, a2, ... an* denote primitive actions and S0 is some beginning situation, such that it denotes the situation resulting from performing action *a1,* followed by *a2,* ...., followed by *an*, beginning in situation S0. For temporal reasoning with situation calculus **[Pinto & Reiter, 93]**, a sequence of ground action terms *a1, a2, a3,......, an* is legal with respect to some theory of action $T_{activity}$ based upon our adoption of the situation calculus as the foundational theory to provide a semantics to the ontology of activity, state and time in TOVE **[Gruninger 97]** iff:-

$T_{activity} \models S0 \leq \mathbf{do([a1, a2, a3, ..., a(n\text{-}1), an], S0)}$

The aspects of the situation calculus and predicates presented in this section will be used later in the chapter in formalizing axioms towards a Theory of Resource Cost Units.

3.4.2 Time Representation in TOVE

Time is represented by points and periods (intervals) on a continuous time line (Fig. 2). A time-point lies within an interval. In every day life, we intuitively associate some degree of uncertainty with the occurrence of an event at a precise time-point. Hence, we associate a time-point as lying within an interval. A time-period is bounded by a start and end time-point. In TOVE, use of Allen's temporal relations [Allen 83] describe the relationships between time-points and/or time-periods. For examples of Allen's temporal relations, refer to Figure 3.

FIGURE 2 Time-Point and Time-Period on Continuous Time Line

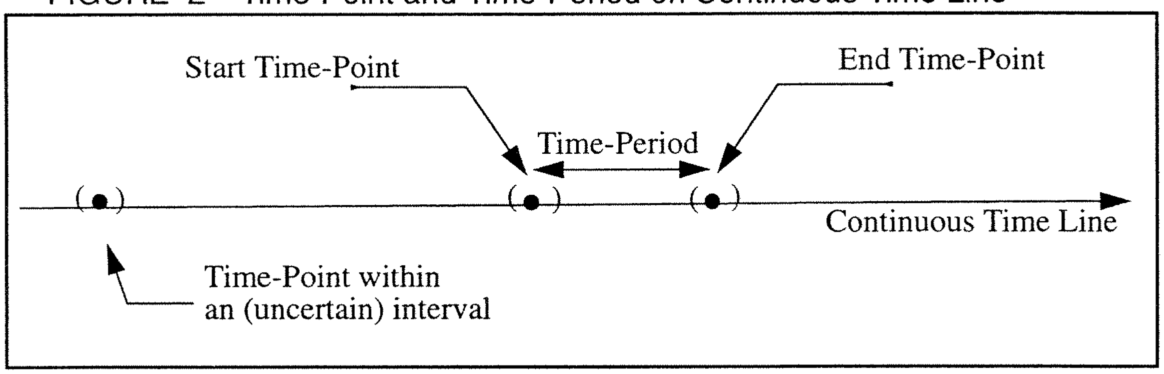
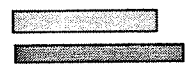
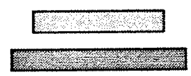


FIGURE 3 Examples of Temporal Relations

Relation	Symbol	Symbol for Inverse	Pictorial Example
X starts Y	s	si	
X before Y	<	>	
X during Y	d	di	
X ends Y	e	ei	

3.4.3 Activity/State Terminology and Semantics

Enterprises are action oriented, and therefore, the ability to represent action lies at the heart of all enterprise models [Tham et al 94]. The CIM-OSA model [ESPRIT 91] stratifies action from the lowest level of a function, to an enterprise activity and up to a business process; the Scheer representation [Scheer 89] defines function specific actions, and the PERA model [William 92] has a two level representation composed of a task at a lower level and a function at the upper level. In the CAM-I cost management system (CMS) model, a function is "a group of activities having a common objective within the business" [Berliner & Brimson 88]. In TOVE, a single entity called an activity spans all of the above. In this section, the terminology and semantics as per TOVE is described.

In TOVE, action is represented by the combination of an activity and its corresponding enabling and caused states [Fox et al 93] [Tham et al 94]. An *activity* is the basic transformational action primitive with which processes and operations can be represented. An enabling state defines what has to be true of the world in order for the activity to be performed. A caused state defines what will be true of the world once the activity has been completed. *An activity along with its enabling and caused states is called an activity-state cluster or simply, activity cluster.*

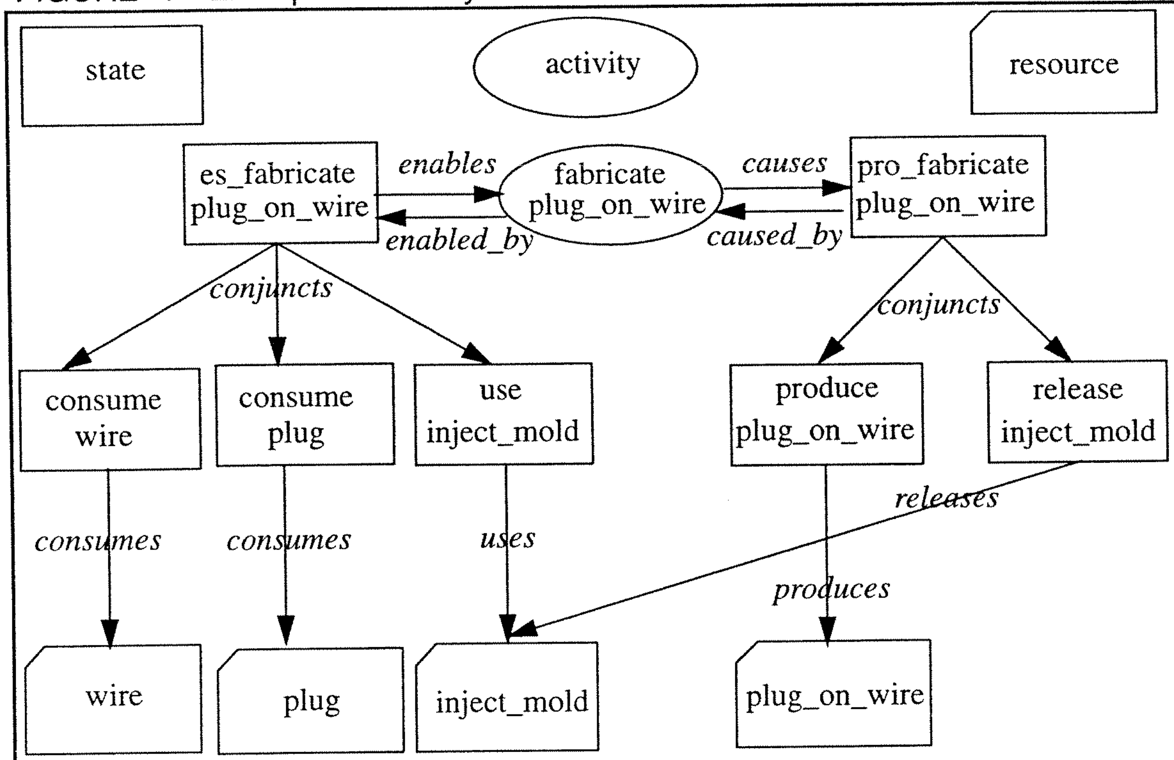
The domain of an activity's status is a set of linguistic constants:-

- **dormant** - the activity may have resources committed to it, the activity is idle and has not been executing before.
- **executing** - the activity is executing.
- **suspended** - the activity was executing and has been forced to an idle state.
- **reExecuting** - the activity is executing again.
- **completed** - the activity has finished.

There are two types of states: *terminal* and *non-terminal.* The *use, consume, release,* and *produce* states are terminal states. Non-terminal states permit us to model some boolean combination of the terminal states.

According to the activity/state, causality, time, and resource ontologies, the usage, consumption, production, or release of a *resource* is represented as a *uses, consumes, produces,* or *releases* relation respectively. These relations - *uses, consumes, produces,* and *releases* - indicate relationships between the entity *state* of the world and a *resource.*

FIGURE 4 Example**: Activity - State Cluster with Resources



** This example illustrates an Object-Oriented Model of Activity, States, Causality, Time, and Resources

The state descriptions of the world can be composed of *conjuncts* (and relations) & *disjuncts* (or relations) to form two composite state descriptions: *enable state* and *caused by state.*

An *enable state* is a description of what is true of the world **before** an activity is performed.

A *caused by state* (inverse of causes) is a description of what is true of the world **after** an activity is performed.

In figure 15, the activity-state cluster indicates that the activity, fabricate plug_on_wire, is enabled by the state tree consisting of the conjunctive non-terminal state, es_fabricate_plug_on_wire, specifying that the terminal states - consume wire, consume plug, use inject_mold - must be satisfied conjunctively for the execution of the activity. Once the activity is completed, it causes what will be true in the real world. This causal effect is shown by the state tree made up of the conjunctive non-terminal state, pro_fabricate plug_on_wire, specifying that the plug_on_wire will be produced and the inject_mold will be released conjunctively on completion of the activity.

3.4.4 Resource and State Terminology and Semantics

"Being a resource" is not an innate property of an object, but is a property that is derived from the role an object plays with respect to an activity [Fadel & Fox, 94].

Hence, the resource ontology includes the concepts of a resource being divisible, quantifiable, consumable, reusable, a component of, committed to, and having usage and consumption specifications.

A state in TOVE represents what has to be true in the world in order for an activity to be performed, or what is true in the world after an activity is completed. States associate resources with activities through the four types of states which reflect the four ways in which a resource is related to an activity - use, consume, release, produce.

The status of a state, and any activity, is dependent on the status of the resources that the activity uses or consumes. All states are assigned a status with respect to a point in time. There are five different status predicates:-

- **committed** - a unit of the resource that the state *consumes or uses* has been reserved for consumption or usage.
- **enabled** - a unit of the resource that the state *consumes or uses* is being consumed or used while the activity is executing.
- **disenabled** - a unit of the resource that the state *consumes or uses* has become unavailable and the activity is suspended
- **reenabled** - a unit of the resource that the state *consumes or uses* is re-available for the activity to resume or reExecute

- **completed** - unit of the resource that the state *consumes or uses* has been consumed or used and is no longer needed.

Resource requirements that enable an activity are specified through the consumption specification (*consume_spec*) and use specification (*use_spec*) of the resources required to enable the activity; whereas, the causal effects of an activity are specified through the produce specification (*produce_spec*) and release specification (*release_spec*) of the resource produced or released on the completion of the activity. A resource may be consumed or used by an enabling state in *continuous_mode* at some rate or discrete mode as some quantity.

For inventory purposes, a resource point (rp) of a resource specifies the quantity of resource at some time point and unit of measure.

3.4.5 Core Terms: Predicates for Activity, State, Resource

The predicates tabulated below are formally defined by Fox et al. [93]. Unless explicitly stated otherwise, all terms defined are fluents whose truth value holds in a situation.

a, a': an activity; s, s': a state description of the world; r: a resource; t, t': time points; q: quantity; rate: rate of consume or use or produce or release state; unit: unit of measure; rp: resource point

TABLE 1 Core Terms: Predicates for Activity, State, Resource, Relations

Activity	*State*	*Resource*	*Relations*
activity (a)	state (s)	resource (r)	enables(s, a)
completed (a)	committed(s, a)	rknown (r)	enabled_by(a, s)
	enabled(s, a)	continuous (r, a)	causes(a, s)
	disenabled(s, a)	discrete (r, a)	caused_by (s, a)
	reenabled(s, a)		
	consume(s, a)	consume_spec(r, a, t, t', q, rate, unit)	consumes(s, r)
	use(s, a)	use_spec(r, a, t, t', q, rate, unit)	uses(s, r)
	produce(s, a)	produce_spec(r, a, t, t', q, rate, unit)	produces(s, r)
	release(s, a)	release_spec(r, a, t, t', q, rate, unit)	releases(s, r)
		rp(r, q, t, unit)	conjuncts(s, s')
			disjuncts (s, s')
			subClass_of(a, a')
			instance_of(a, a')
			has_subactivity(a, a')

3.5 Concluding Remarks

This chapter has laid the foundation for one to understand the ontological engineering methodology deployed in designing a Cost Ontology for Enterprise Modelling and a Micro-theory of ABC in TOVE, both of which are presented in the next chapter.

The understanding of answering competency questions through our Cost Ontology and Micro-theory for ABC form the basis of applying the developments of this research in the companies of our corporate partners, deHavilland and BHP. This is done in Chapters 6 and 7 respectively. Understanding the concept of Re-usability and Reducibility form the theme for the testing of the generality criteria of the Cost Ontology being met with reference to a popular ABC software - Netprohet II - in Ch. 8.

The predicates presented in this chapter have all been defined in the references cited. These predicates will be extensively used in formalizing several axioms in FOL for the remainder of this report.

The understanding of the states of a resource as being committed, enabled, disenabled, reenbled, and completed form the basis of our having to associate four unit resource costs to each resource. The statii of an activity as being dormant, executing, suspended, reExecuting and completed pave the way towards our understanding of activity costs in a dynamically changing world. In short, this chapter lays the foundation for the understanding of the temporal cost behaviour for resource and activity.

CHAPTER 4 Cost Ontology for Enterprise Modelling

4.1 Introduction

The deployment of ABC to improve business is commonly referred to as Activity-Based Management (ABM). The motivation for this chapter is to develop a basic ontology of costs to support Activity-Based Management (ABM). Therefore, to provide direction to the design and development of the cost ontology, a set of informal questions in the domain of ABM are posed, so that the cost ontology developed is sufficiently competent to provide answers to them. The Cost Ontology designed lays the foundation for a Micro-theory of ABC. The Cost Ontology and Micro-theory of ABC provide a Formalization of the ABC Principle.

This chapter has been based upon [Tham et al. 94].

4.2 Competency Questions for the Cost Ontology

The fundamental and commonly asked informal queries in the domain of ABM that need to be readily answered by the enterprise are:

1. What is the instantaneous and cumulative cost of a resource used in an activity a at time t?

2. What is the instantaneous and cumulative cost of an activity a at time t?

3. What is the instantaneous and cumulative cost of cost object o at time t?

4. What is the cost of resource at a point in time when it is being consumed or used at some continuous rate of consumption or usage?

5. What is the cost of the resource if it is required as some discrete or batch quantity by an activity?

6. What is the cost effect on a resource, given that the condition or state of the resource changes with time?

7. What cost effect does the state of a resource have upon the cost of an activity?

8. What cost effect does the state of a resource have upon the cost of a cost object?

9. What is the cumulative cost of the class of activities a?

10. What is the cumulative cost of the class of orders o?

11. What is the cost of each subClass activity, a_i, when a_i has activity instances, a_{ij}'s?

12. What is the cost of Class_ Activity, a'_{ix}, (e.g., the class activity, Distribution) given that a'_{ix} (viz., Distribution) has subClass activities, a_i's? (e.g., Order-Picking, Palletizing, Material Handling, Shipping)? [Note: the subClass activity, Material Handling, may have activity instances like Hand Pallet Truck Handling, Fork-Lift Truck Handling, and Conveyorized Handling].

13. What is the cost of Cost_Order_Class, x_c, (e.g., x_c may include all instances of orders that is fulfilled for the export sector or the electronic industry sector) given that x_c includes cost order instances, x_{ci}'s?

The above questions are by no means an exhaustive set of queries, but they provide direction to the design and development of the cost ontology. We term such questions as competency questions as the extent to which the cost ontology is able to provide answers to these queries with minimal ambiguity and maximum precision will be a reflection of the competency of the cost ontology developed [Fox et al. 93].

Given complete knowledge of time, activities, status of states, resources and primitive resource costs required by activities (Closure Axioms formalized later), the design of Cost Ontology provides solutions to the specific competency questions. These solutions are pursued and deduced from an ABC perspective.

4.3 Cost Ontology for TOVE

> Definition: Cost is that entity which represents the temporal fiscal or monetary dimension, attribute, or characteristic of an enterprise activity, and may be referred to as *activity cost.*

In TOVE, costs will only change whenever the statii of states, (hence resources), and activities change. Hence, we consider a cost event occurring when the status value of a state and activity change. Cost is treated constant with time until a status change occurs. In fact, we are applying situational calculus to model costs in a continuous dynamically changing world through the representation of discrete number of states of the world.

In order to reason about activities and their costs, we require a precise representation of the following:-

1. the generic classification or taxonomies of activity costs that intuitively or rationally present themselves in common sense enterprise modelling and our cost management perspective with TOVE;
2. the computational aspects in quantifying the types of activity costs.

Towards this end, the cost ontology for TOVE will serve as the data dictionary for the discourse domain of the cost advisor - an agent to communicate cost related data, to perform and to deduce cost computations that assist a human decision maker to accomplish ABC management at the generic level of enterprise modelling. The cost ontology uses the terminology and semantics of activity, state, resource and time that have been defined at a generic level for the TOVE environment [TOVE 93].

4.4 Cost Point of a Resource for an Activity

ABC assigns costs to activities based on their use of resources. Therefore, our design of the cost ontology begins from the resource end.

The quantification of activity cost or the cost value of an activity associated with a required resource in TOVE is specified through the usage of the resource cost point predicate.

> Definition: The *resource cost point* predicate, **cpr(a, c, t, r),** specifies the cost value, c, (monetary units) of a resource, r, required by an activity, a, up to time point, t.

If a resource of the terminal use or consume state, s, for activity, a, is enabled at time point, t, implies there exist a cost value, c, at time t, for activity, a, that uses or consumes r. The time interval, $[t_s, t_e]$, during which a resource is used or consumed by an activity is specified in the use or consume specification of the resource [Fadel et al. 94]. Hence,

$$\textit{Axiom: } \forall a, r, q, t_s, t_e, rate, unit, t, t_s \leq t < t_e, (use\_spec\,(r, a, t_s, t_e, q, rate, unit) \vee (consume\_spec\,(r, a, t_s, t_e, q, rate, unit) \supset \exists c, cpr\,(a, c, t, r) \qquad \text{(FOL 3)}$$

[The computation for the resource cost point is discussed later in Axiom (FOL 35)].

Example: cpr (assemble_clip_reading_lamp, 120, 75, hex_nut)

The above example indicates the resource_cost_point for the activity, clip_reading_lamp_assembly, is of cost_value 120 monetary units at time point 75, for the resource identified as hex_nut.

4.5 Cost Point of Activity

Definition: The *cost point of activity* predicate, cpa, defined as **cpa(a, c, t)**, specifies the aggregate cost_value, c, of the activity, a, at time_ point, t, given that the activity, a, uses and/or consumes one or more resources up to time t.

The cost value argument of the cpa predicate is obtained by the summation of the cost value argument c_i for each resource cost point predicate, cpr, for all resources used and/or consumed by the activity, a, up to time t.

By definition, the cost point of activity at time t returns the summation of all cost values c_i over all resources required by the activity up to time t as indicated in the axiom schema:-

Axiom (schema): For each activity, a, and $\forall\, c_1, c_2, c_3, \ldots, c_n, r_1, r_2, \ldots r_n, t, \exists\, c,$ *cpa*

$$(a, c, t) \equiv cpr\,(a, c_1, t, r_1) \wedge cpr\,(a, c_2, t, r_2) \wedge cpr\,(a, c_3, t, r_3) \wedge \ldots \wedge cpr\,(a, c_n, t, r_n)$$

$$\wedge\, c = c_1 + c_2 + \ldots + c_n \qquad \text{(FOL 4)}$$

[The computation of the cost point of activity is discussed later in the chapter].

4.6 Taxonomy of Resource Cost Units

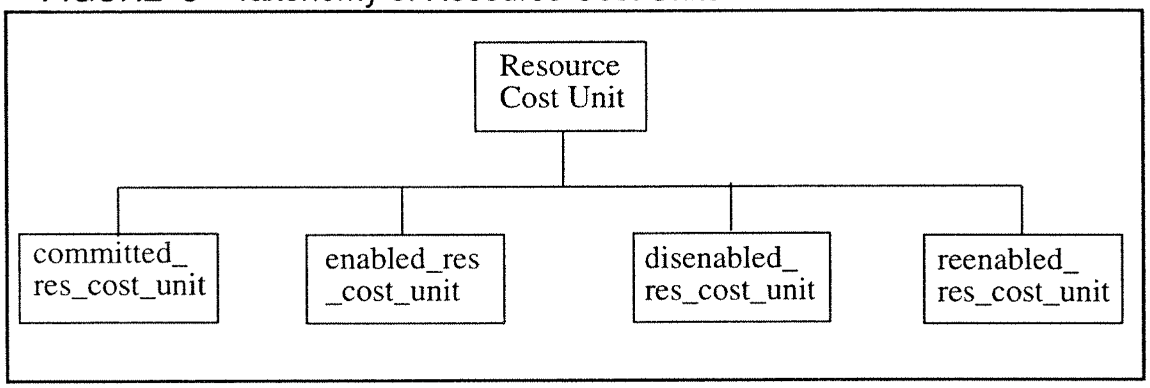
Since ABC assigns costs to activities based on their use of resources, the logical formulation of ABC must be premised on the existence of some given or identifiable cost of 1 unit amount of resource.

Definition: In TOVE, we define the basic or primitive cost value of 1 unit amount of a resource consumed or used by an activity as the *resource cost unit* of the resource for the activity.

The status of an activity depends on the status of the resources required by the activity. Depending upon the use and/or consume terminal states of the enabling states of an activity, the activity status may be dormant, executing, suspended, reExecuting, or completed. Before being completed, an activity status may iterate through the various status values.

Since the status of a resource may be committed, enabled, disenabled or reenabled, the Taxonomy of Resource Cost Units consists of committed_res_cost_unit, enabled_res_cost_unit, disenabled_res_cost_unit and reenabled_res_cost_unit.

FIGURE 5 Taxonomy of Resource Cost Units



Definitions pertaining to the resource cost unit taxonomy are as follows:-

- the committed_res_cost_unit predicate defined as **committed_res_cost_unit (a, r, q, v_1)**, provides the cost linkage between an activity and a resource committed to that activity, through its cost metric arity, v_1, expressed as $/unit of resource;
- the enabled_res_cost_unit predicate defined as **enabled_res_cost_unit (a, r, q, v_2)**, provides the cost linkage between an activity and a resource that enables that activity to execute, through its cost metric arity, v_2, expressed as $/unit of resource;
- the disenabled_res_cost_unit predicate defined as **disenabled_res_cost_unit (a, r, q, v_3)**, provides the cost linkage between an activity and a disenabled resource that suspends the activity's execution, through its cost metric arity, v_3, expressed as $/ unit of resource;

- the reenabled_res_cost_unit predicate defined as **reenabled_res_cost_unit (a, r, q, v_4)**, provides the cost linkage between an activity and a reenabled resource that enables the activity to resume execution, through its cost metric arity, v_4, expressed as $/unit of resource.

We consider the resource cost unit primitives, viz., v_1, v_2, v_3, v_4, as being the four cost attributes of the use and consume states associated with each resource specified for an activity.These resource cost units must be given or identifiable in the enterprise data model. Since ABC assigns costs to activities based on their use of resources, the logical formulation of ABC must be premised on the existence and knowledge of the resource cost units. Closure for the activity cost management domain must be ensured by the complete knowledge (completeness) of time, activities, status of states, resources required for activities together with the resource cost units (see Closure Axioms at the end of this chapter).

From an ABC perspective, incompleteness would make it impossible to compute the costs of activities if resources and resource cost units required by activities are not identified. Consequently, the assignment of activity costs to "cost objects" cannot be made. Incompleteness would lead to ambiguity for activity cost computations in the domain of activity based cost management.

Initiated by the competency questions, the design, development and formalization of the Cost Ontology in first order logic identifies that integrated ontologies for activity, resource and time are required to achieve completeness for ABM.

4.7 Mapping: Cost Object in ABC with Cost Order in TOVE

Within the ABC literature, activities are performed for reasons that are termed "cost objects" [Turney 92]. These reasons or cost objects are satisfied through the completion of activities; and may be perceived as being analogous to the concept of job orders that must be satisfied by the completion of activities that require resources.

Within the TOVE enterprise modelling paradigm, there are four generic and identifiable types of orders for which activities are performed at any given time point t. An activity is performed for a specific customer order, an internal order, a forecast order, or a purchase order. In other words, an activity, a, at any time point, t, bears a relationship, viz., *has_order,* with the object, x, where x may be a customer order, an internal order, a forecast order, or a purchase order [TOVE 93].

Intuitively, there is direct mapping between the concept of order in TOVE to the concept of cost object in ABC. Therefore, from a cost perspective, the concept of order in TOVE is associated with the term cost_order in the Cost Ontology to mean cost object in ABC.

We state the following axiom:-

The order of an activity is the cost order of that activity.

Axiom: $\forall a, x, t, activity(a) \wedge order(x) \wedge has\_order(a, x, t) \equiv cost\_order(x, a, t)$ (FOL 5)

4.8 Activity and Resource Cost Taxonomy

Our terminology and semantics for the knowledge representation of activity, state, resource and time explicitly recognize the temporal status of the states associated with the resources required by an activity, which in turn affect the status of the activity.

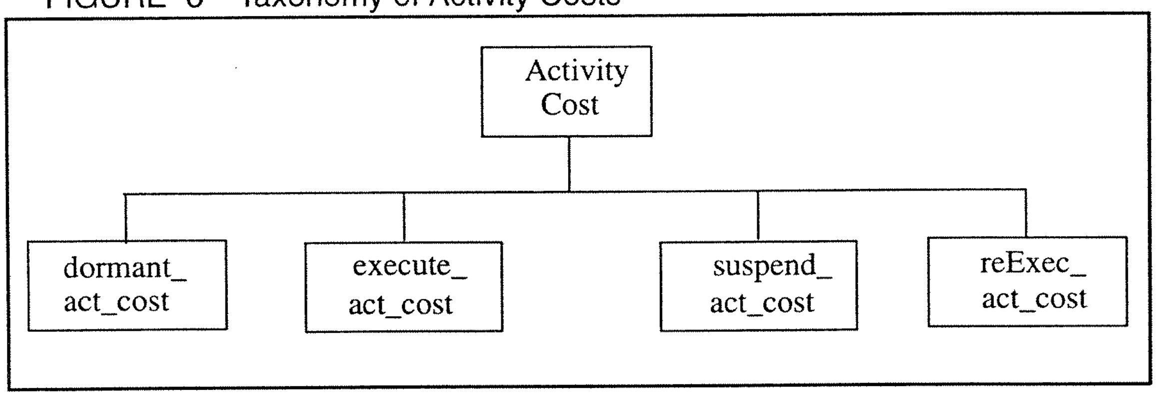
The ABC concept includes the assignment of costs to activities based on their use of resources. Therefore, to integrate the cost ontology with TOVE's activity, state, resource and time ontologies from an ABC perspective, the cost ontology considers the temporal behaviour of an activity and the cost of the activity that are closely associated with the status value of the enabling states and activity, together with the cost attributes of the use and consume terminal states in the enabling state tree for an activity.

Before an activity is completed, it is quite possible that the activity status value may have cycled through the dormant, executing, suspended and reExecuting status values. Hence, considering the taxonomy of resource cost units, and the temporal nature of activity status values, our activity cost profile must capture the cost of performing the activity dependent on its status value and resource cost unit allocation.

In TOVE, explicit recognition is given to the temporal behaviour of activity cost by defining four primitive types of activity cost predicates based on the status of the activity which depends on the status of the resources required by the activity. To be consistent and complete with the changing status of the activity, the temporal behaviour of costs are captured through four *activity cost predicates* specified as follow (refer to Taxonomy of Activity Costs):-

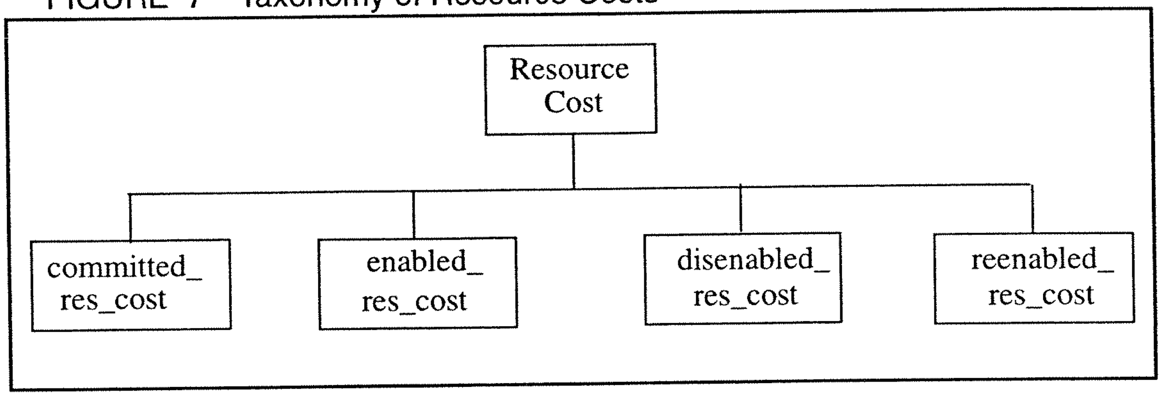
1. **dormant_act_cost (a, c, t)** associated with dormant activity cost,
2. **execute_act_cost (a, c, t)** associated with executing activity cost,
3. **suspend_act_cost (a, c, t)** associated with suspended activity cost, and
4. **reExec_act_cost (a, c, t)** associated with reExecuting activity cost.

FIGURE 6 Taxonomy of Activity Costs



The status of the activity depends on the status of the resources required by the activity. Therefore, from an ABC perspective, we assign costs to each of the above four primitive types of activity costs by considering the status of the states associated with the resources and the corresponding resource costs. Hence, in TOVE, to assign costs to the four primitive types of activity costs, we also give explicit recognition to the computation of four primitive types of resource costs based on the status of the states associated with the resource. This is achieved through four *resource cost predicates* specified as follows (refer to Taxonomy of Resource Costs):-

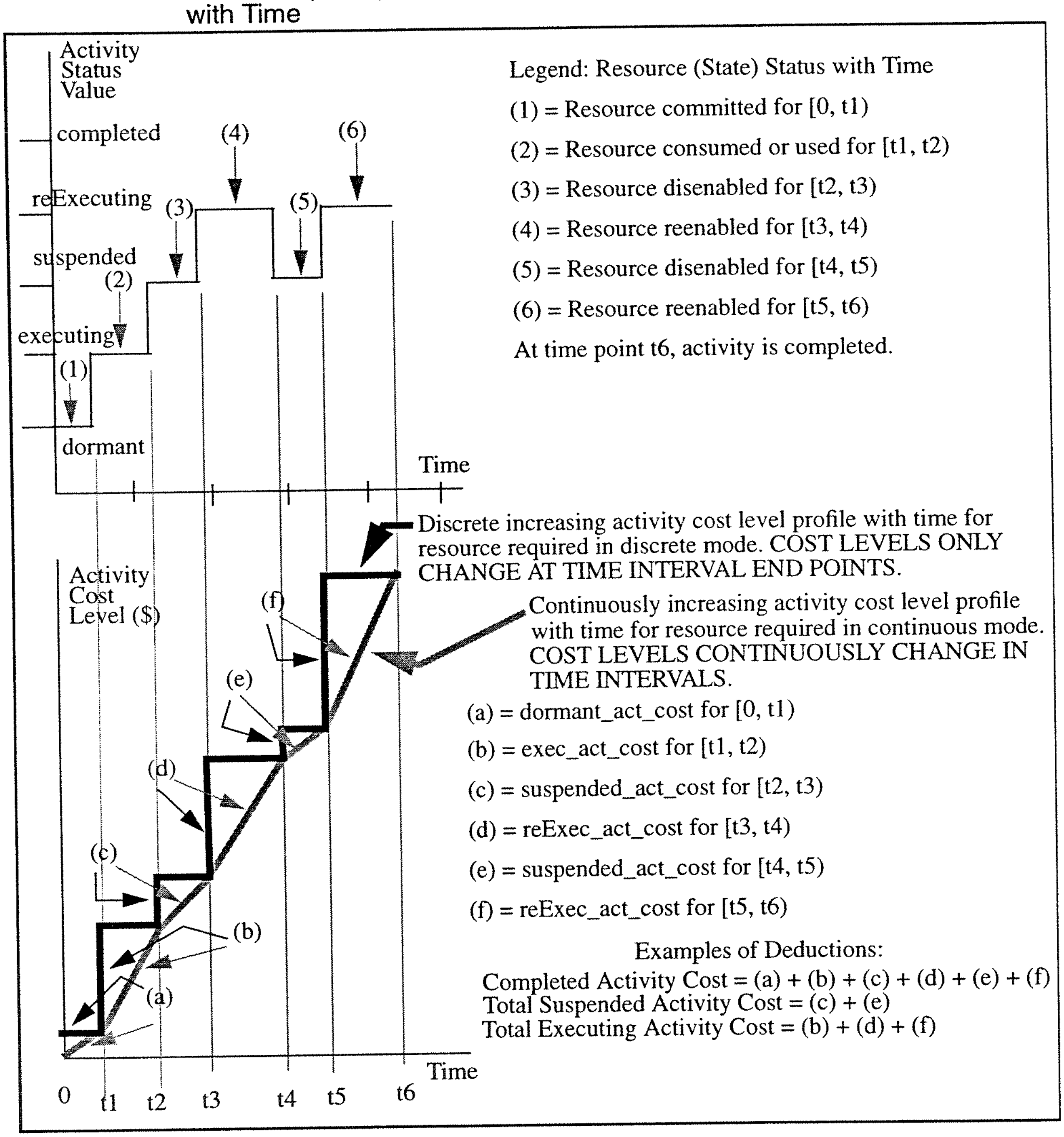
FIGURE 7 Taxonomy of Resource Costs



1. **committed_res_cost (a, r, c, t)** for committed resource cost,
2. **enabled_res_cost (a, r, c, t)** for enabled resource cost,
3. **disenabled_res_cost (a, r, c, t)** for disenabled resource cost, and
4. **reenabled_res_cost (a, r, c, t)** for reenabled resource cost.

In general, from the activity status and activity cost level profile (Fig. 21), an activity holds it status value for a time-period or time-interval with start point t and end point t'. During the interval [t_s, t_e], the activity requires resource r. The cost allocation, c, for an activity that requires the resource on a continuous basis during that interval is computed as the resource cost unit value, v, multiplied by the length of the time interval, (t_e - t_s), for which the activity holds a steady status value, multiplied by the **rate** of resource as specified in the **use_spec (r, a, t_s, t_e, q, rate, unit)** or **consume_spec (r, a, t_s, t_e, q, rate, unit)** respectively. However, the cost allocation, c, for an activity that requires the resource on a discrete basis during that interval is computed as the resource cost unit value, v, multiplied by the quantity, **q**, of the resource as specified in the **use_spec (r, a, t_s, t_e, q, rate, unit)** or **consume_spec (r, a, t_s, t_e, q, rate, unit)** respectively.

FIGURE 8 Resource (State) Status, Activity Status and Activity Cost Profile with Time



4.9 Resource Costs

The ABC perspective demands that the assignment of costs to an activity be based on the costs of the resources required by the activity.

The classification of a resource cost is based on the status value of the state associated with a resource required by an activity during a time interval. In computing the resource cost for this time interval, the total quantity of the resource required by the activity for the time interval must be considered. Since resource costs may be demanded at any point in time based on our competency questions, the computation of resource costs are established at the end points of a time interval, at time points during the interval, and at some future time point given that we have some partial information of resource costs up to a current time point. Further, owing to the temporal and repetitive occurrences of the resource (state) status, the aggregation of resource costs must be established over multiple intervals.

Hence, a formalization of resource cost of an activity include the axioms for:-

- definitions of status intervals of a (resource) state,
- resource cost points of a single status interval,
- resource costs with partial information, and
- the aggregation of resource costs over multiple intervals.

4.9.1 Definitions of Status Intervals of a (Resource) State

Definition: *Committed (status) Interval*: The committed interval [t, t'), of a resource, r, associated with activity, a, implies that the committed status of the state associated with activity, a, holds for all time points from t onwards up to and excluding t'.

The axiom for the committed interval is:

Axiom: $\forall a, s, r, t, t', committed\_interval(a, r, t, t') \equiv \forall t'', t \leq t'' < t' \supset holdsT(committed(s, a), t'') \wedge \neg holdsT(committed(s, a), t') \wedge holds(committed(s, a), t)$ (FOL 6)

Definition: *Enabled (status) Interval*: The enabled interval [t, t'), of a resource, r, associated with activity, a, implies that the enabled status of the state associated with activity, a, holds for all time points from t onwards up to and excluding t'.

The axiom for the enabled interval is:

$$\textbf{Axiom: } \forall a, s, r, t, t', \textit{enabled\_interval}(a, r, t, t') \equiv \forall t'', t \le t'' < t' \supset \textit{holdsT}(\textit{enabled}(s, a), t'') \wedge \neg \textit{holdsT}(\textit{enabled}(s, a), t') \wedge \textit{holdsT}(\textit{enabled}(s, a), t) \quad \text{(FOL 7)}$$

Definition: *Disenabled (status) Interval*: The disenabled interval [t, t'), of a resource, r, associated with activity, a, implies that the enabled status of the state associated with activity, a, holds for all time points from t onwards up to and excluding t'.

The axiom for the disenabled interval is:

$$\textbf{Axiom: } \forall a, s, r, t, t', \textit{disenabled\_interval}(a, r, t, t') \equiv \forall t'', t \le t'' < t' \supset \textit{holdsT}(\textit{disenabled}(s,a), t'') \wedge \neg \textit{holdsT}(\textit{disenabled}(s,a), t') \wedge \textit{holdsT}(\textit{disenabled}(s,a), t) \quad \text{(FOL 8)}$$

Definition: Reenabled (status) Interval: The reenabled interval (t, t'), of a resource, r, associated with activity, a, implies that the reenabled status of the state associated with activity, a, holds for all time points from t onwards up to and excluding t'.

The axiom for the reenabled interval is:

$$\textbf{Axiom: } \forall a, s, r, t, t', \textit{reenabled\_interval}(a, r, t, t') \equiv \forall t'', t \le t'' < t' \supset \textit{holdsT}(\textit{reenabled}(s, a), t'') \wedge \neg \textit{holdsT}(\textit{reenabled}(s, a), t') \wedge \textit{holdsT}(\textit{reenabled}(s, a), t) \quad \text{(FOL 9)}$$

4.9.2 Resource Costs Points of a Single Status Interval

A resource may be committed, enabled, disenabled or reenabled for a time interval. Resource cost points must be computed for each time point during that time interval.

The committed_res_cost, enabled_res_cost, disenabled_res_cost and reenabled_res_cost must take into consideration the mode, viz., continuous mode, [**continuous_mode (r, a)**] or discrete mode, [**discrete_mode (r, a)**], in which the resource is required during a time interval. The costs for a resource required in continuous mode by an activity is a function of the resource cost unit value, the duration of the time period, and the rate object of the use_spec or consume_spec predicate. The costs for a resource required in discrete mode by an activity is a function of the resource cost unit value, and the quantity (q) object of the use_spec or consume_ spec predicate.

For time points that are strictly between the end points of a single status interval, given that a resource is used or consumed by an activity on a continuous time basis, the resource cost point of the activity has a value which is a function of the time interval and the quantity of resource required by the activity. However, if the resource is required in some fixed discrete quantity only, then the resource cost is a function of the discrete resource quantity only.

> For a committed state resource required by an activity in *continuous* mode over a time interval [t, t'] and [$t \leq t'' \leq t'$], we state the following axiom:- [Note: the cost of capital "tied in" to the committed resource is not considered in this chapter. However, a more detailed analysis is presented in Chapter 5.]
>
> Axiom: Assuming that the cost of capital for the resource committed is not considered, and given that the resource is required at a continuous ***rate***, then, for all time points during the commit interval of a resource, the cost of committed resource is equivalent to [its commit resource cost unit *(v)*] x [***rate*** of consumption or usage] x [the duration of the time interval for which the resource is committed]. We state this axiom in FOL as follows:-

Axiom: $\forall a, r, c, v, t, t', t'', t \leq t'' \leq t'$, ***committed_interval (a, r, t, t')*** $\wedge$ ***continuous_mode (r, a)*** $\wedge$ ***[use_spec (r, a, t, t', q, rate, unit)*** $\vee$ ***consume_spec (r, a, t, t', q, rate, unit)]*** $\supset$ ***[committed_res_cost (a, r, c, t'')*** $\equiv$ ***committed_res_cost_unit (a, r, q, v)*** $\wedge$ ***c = v.(t'' - t).rate]*** **(FOL 10)**

For a committed state resource required by an activity in *discrete* mode over a time interval [t, t'] and [$t \leq t'' \leq t'$], we state the following axiom:- [Note: the cost of capital "tied in" to the committed resource is not considered in this chapter. However, a more detailed analysis is presented in Chapter 5.]

Axiom: Assuming that the cost of capital for the resource committed is not considered, and given that the resource is required as a discrete quantity q, then, for all time points during the commit interval of a resource, the cost of committed resource is equivalent to [its commit resource cost unit *(v)*] x [discrete quantity *q* required]. This axiom stated in FOL is as follows:-

Axiom: $\forall a, r, c, v, t, t', t \leq t'' \leq t'$, ***committed_interval (a, r, t, t')*** $\wedge$ ***discrete_mode (r, a)*** $\wedge$ ***[use_spec (r, a, t, t', q, rate, unit)*** $\vee$ ***consume_spec (r, a, t, t', q, rate, unit)]*** $\supset$ ***[committed_res_cost (a, r, c, t'')*** $\equiv$ ***committed_res_cost_unit (a, r, q, v)*** $\wedge$ ***c = v.q]*** **(FOL 11)**

For an enabled state resource required by an activity on a *continuous* basis over a time interval [t, t'] and [$t \leq t'' \leq t'$], we state the following axiom:-

Axiom: Given that a resource is required by an activity at a *continuous* rate, and the resource is in an enabled state for a period [t, t'], then for all time points during [t, t'], the cost of the enabled resource is equivalent to [its enabled resource cost unit (*v*)] x [rate at which it is consumed or used] x [duration of time interval for which the resource is enabled]. We state this axiom in FOL as follows:-

Axiom: $\forall a, r, c, v, t, t', t'', t \leq t'' \leq t'$, ***enabled_interval (a, r, t, t')*** $\wedge$ ***continuous_mode (r, a)*** $\wedge$ ***[use_spec (r, a, t, t', q, rate, unit)*** $\vee$ ***consume_spec (r, a, t, t', q, rate, unit)]*** $\supset$ ***[enabled_res_cost(a,r,c,t'')*** $\equiv$ ***enabled_res_cost_unit(a,r,q,v)*** $\wedge$ ***c=v.(t''-t).rate]*** **(FOL 12)**

For an enabled state resource required by an activity on a *discrete* basis over a time interval [t, t'] and [$t \le t'' \le t'$], we state the following axiom:-

Axiom: Given that a discrete quantity of resource is required by an activity, and the resource is in an enabled state for a period [t, t'], then for all time points during [t, t'], the cost of the enabled resource is equivalent to [its enabled resource cost unit (*v*)] x [quantity q of resource required]. We state this axiom in FOL as follows:-

Axiom: ***∀a, r, c, v, t, t', t ≤ t'' ≤ t', enabled_interval (a, r, t, t') ∧ discrete_mode (r, a) ∧ [use_spec (r, a, t, t', q, rate, unit) ∨ consume_spec (r, a, t, t', q, rate, unit)] ⊃ [enabled_res_cost (a, r, c, t'') ≡ enabled_res_cost_unit (a, r, q, v) ∧ c = v.q]*** (FOL 13)

For a disenabled state resource required by an activity on a *continuous* basis over a time interval [t, t'] and [$t \le t'' \le t'$], we state the following axiom:- [Note: the "lost opportunity cost" due to a resource being disenabled is not considered in this chapter. However, a more detailed analysis is presented in Chapter 5.]

Axiom: Assuming that "lost opportunity cost" due to a resource being disenabled is not considered, and given that the resource is required at a continuous ***rate***, then, for all time points during the disenabled interval of a resource, the cost of the disenabled resource is equivalent to [its disenabled resource cost unit ***(v)***] x [***rate*** of consumption or usage] x [the duration of the time interval for which the resource is disenabled]. We state this axiom in FOL as follows:-

Axiom:∀a, r, c, t, t', t'', t ≤ t'' ≤ t', disenabled_interval (a, r, t, t') ∧ continuous_mode (r, a) ∧ [use_spec (r, a, t, t', q, rate, unit) ∨ consume_spec (r, a, t, t', q, rate, unit)] ⊃ [disenabled_res_cost (a, r, c, t'') ≡ disenabled_res_cost_unit (a, r, q, v) ∧ c = v.(t'' - t) .rate] (FOL 14)

For a disenabled state resource required by an activity on a discrete basis over a time interval [t, t'] and [$t \le t'' \le t'$], we state the following axiom:- [Note: the "lost opportunity cost" due to a resource being disenabled is not considered in this chapter. However, a more detailed analysis is presented in Chapter 5.]

Axiom: Assuming that "lost opportunity cost" due to a resource being disenabled is not considered, and given that the resource is required in a discrete quantity q, then, for all time points during the disenabled interval of a resource, the cost of the disenabled resource is equivalent to [its disenabled resource cost unit *(v)*] x [discrete quantity q required]. We state this axiom in FOL as follows:-

$$\textbf{Axiom}: \forall a, r, c, t, t', t \le t'' \le t', disenabled\_interval(a, r, t, t') \wedge discrete\_mode(r, a) \wedge [use\_spec(r, a, t, t', q, rate, unit) \vee consume\_spec(r, a, t, t', q, rate, unit)] \supset [disenabled\_res\_cost(a, r, c, t'') \equiv disenabled\_res\_cost\_unit(a, r, q, v) \wedge c = v.q] \quad \text{(FOL 15)}$$

For a reenabled state resource required by an activity on a *continuous* basis over a time interval [t, t'] and [$t \le t'' \le t'$], we state the following axiom:-

Axiom: Given that a resource is required by an activity at a *continuous* rate, and the resource is in a reenabled state for a period [t, t'], then for all time points during [t, t'], the cost of the reenabled resource is equivalent to [its reenabled resource cost unit (*v*)] x [rate at which it is consumed or used] x [duration of time interval for which the resource is reenabled]. We state this axiom in FOL as follows:-

$$\textbf{Axiom}: \forall a, r, c, t, t', t'', t \le t'' < t', reenabled\_interval(a, r, t, t') \wedge continuous\_mode(r, a) \wedge [use\_spec(r, a, t, t', q, rate, unit) \vee consume\_spec(r, a, t, t', q, rate, unit)] \supset [reenabled\_res\_cost(a, r, c, t'') \equiv reenabled\_res\_cost\_unit(a, r, q, v) \wedge c = v.(t'' - t).rate] \quad \text{(FOL 16)}$$

For a reenabled state resource required by an activity on a discrete basis over a time interval [t, t'] and [$t \le t'' \le t'$], we state the following axiom:-

Axiom: Given that a discrete quantity q of resource is required by an activity, and the resource is in an reenabled state for a period [t, t'], then for all time points during [t, t'], the cost of the reenabled resource is equivalent to [its reenabled resource cost unit (*v*)] x [discrete quantity *q* required]. We state this axiom in FOL as follows:-

Axiom: $\forall a, r, c, t, t', t \leq t'' \leq t'$, ***reenabled_interval (a, r, t, t')*** $\wedge$ ***discrete_mode(r, a)*** $\wedge$ ***[use_spec (r, a, t, t', q, rate, unit)*** $\vee$ ***consume_spec (r, a, t, t', q, rate, unit)]*** $\supset$ ***[reenabled_res_cost (a, r, c, t')*** $\equiv$ ***reenabled_res_cost_unit (a, r, q, v)*** $\wedge$ $c = v.q]$ **(FOL 17)**

4.9.3 Resource Costs with Partial Information

One of the demands put forth by our competency questions is the requirement to compute the cumulative resource costs of an activity up to some future time point, t', given that we are at the present time point, t.

The time interval of interest for this situation is (t, t'). We only have partial information of the time interval, (t, t'), as we only know the cost value, c, at time t without knowing all actions that occur between t and t'. Hence, if resource costs are to be computed with partial knowledge, we need to explicitly express that no action occurred between t and t' that could change the status for time points beyond t. Hence, given that we have partial information of the resource cost point at present time t, my axioms pertaining to resource cost points for all t' > t make use of the predicate occursBet.

Definition: The predicate **occursBet** is used to represent the fact that an action occurs between two time points t and t'.

Axioms 18 through 25 make use of the negation of the predicate occursBet (denoted as $\neg$ ***occursBet***) to expressly state that, if no action occurs between t and t', and the cost of resource at t has value c, then the cost of the resource at t' also has value c since no action has occurred between t and t'.

If at time t the committed/enabled/disenabled/reenabled resource cost is of value c, and the status of the state for the activity is committed/enabled/disenabled/reenabled at time t', and the status of the state remains the same respectively between t and t', then the resource cost value, c', of the activity at t' is the closest respective resource cost value c of time t plus the cost of the resource used or consumed during the period (t' - t). This is formalized in Axioms (FOL 18) through (FOL 25) for resources that are required on a continuous or a discrete basis over a time interval. These axioms also formalize the procedure of computing cost of a resource used or consumed at t' as a cumulative cost up to t', and as an increasing function of the time period between t and t'.

In short, if the cost of resource used or consumed is c at time point t, the cost of the resource at some future time point t' will be [c + (cost of resource used or consumed from t to t')].

For a committed state resource required by an activity on a continuous basis over a time interval [t, t'], we state the following FOL axiom:- [Note: the cost of capital "tied in" to the committed resource is not considered in this chapter. However, a more detailed analysis is presented in Chapter 5.]

Axiom: $\forall$***a, r, c, c', t, t', committed_res_cost (a, r, c, t)*** $\wedge$ ***continuous_mode (r, a)*** $\wedge$ ***[use_spec (r, a, t, t', q, rate, unit)*** $\vee$ ***consume_spec (r, a, t, t', q, rate, unit)]*** $\wedge$ ***status (s, a, committed, t')*** $\wedge \neg$ ***occursBet (commit(s,a), t, t')*** $\supset$ ***[committed_res_cost (a, r, c', t')*** $\equiv$ ***committed_res_cost_unit (a, r, q, v)*** $\wedge$ ***c' = c + v.(t' - t).rate]*** **(FOL 18)**

For a committed state resource required by an activity on a discrete basis over a time interval [t, t'], we state the following FOL axiom:- [Note: the cost of capital "tied in" to the committed resource is not considered in this chapter. However, a more detailed analysis is presented in Chapter 5.]

Axiom: $\forall$***a, r, c, c', t, t', committed_res_cost (a, r, c, t)*** $\wedge$ ***discrete_mode (r, a)*** $\wedge$ ***[use_spec (r, a, t, t', q, rate, unit)*** $\vee$ ***consume_spec (r, a, t, t', q, rate, unit)]*** $\wedge$ ***status (s, a, committed, t')*** $\wedge \neg$ ***occursBet (commit (s, a), t, t')*** $\supset$ ***[committed_res_cost (a, r, c', t')*** $\equiv$ ***committed_res_cost_unit (a, r, q, v)*** $\wedge$ ***c' = c + v.q]*** **(FOL 19)**

For an enabled state resource required by an activity on a continuous basis over a time interval [t, t'], we state the following FOL axiom:-

$$\textit{Axiom: } \forall a, r, c, c', t, t', enabled\_res\_cost(a, r, c, t) \wedge continuous\_mode(r, a) \wedge [use\_spec(r, a, t, t', q, rate, unit) \vee consume\_spec(r, a, t, t', q, rate, unit)] \wedge status(s, a, enabled, t') \wedge \neg occursBet(enabled(s, a), t, t') \supset [enabled\_res\_cost(a,r,c',t') \equiv enabled\_res\_cost\_unit(a,r,q,v) \wedge c' = c + v.(t'-t).rate] \quad \text{(FOL 20)}$$

For an enabled state resource required by an activity on a discrete basis over a time interval [t, t], we state the following FOL axiom:-

$$\textit{Axiom: } \forall a, r, c, c', t, t', enabled\_res\_cost(a, r, c, t) \wedge discrete\_mode(r, a) \wedge [use\_spec(r, a, t, t', q, rate, unit) \vee consume\_spec(r, a, t, t', q, rate, unit)] \wedge status(s,a,enabled,t') \wedge \neg occursBet(enabled(s,a), t, t') \supset [enabled\_res\_cost(a, r, c', t') \equiv enabled\_res\_cost\_unit(a, r, q, v) \wedge c' = c + v.q] \quad \text{(FOL 21)}$$

For a disenabled state resource required by an activity on a continuous basis over a time interval [t, t'], we state the following FOL axiom:- [Note: the "lost opportunity cost" due to a resource being disenabled is not considered in this chapter. However, a more detailed analysis is presented in Chapter 5.]

$$\textit{Axiom: } \forall a, r, c, c', t, t', disenabled\_res\_cost(a, r, c, t) \wedge continuous\_mode(r, a) \wedge [use\_spec(r, a, t, t', q, rate, unit) \vee consume\_spec(r, a, t, t', q, rate, unit)] \wedge status(s, a, disenabled, t') \wedge \neg occursBet(disenabled(s,a), t, t') \wedge \supset [disenabled\_res\_cost(a, r, c', t') \equiv disenabled\_res\_cost\_unit(a, r, q, v) \wedge c' = c + v.(t' - t).rate] \quad \text{(FOL 22)}$$

For a disenabled state resource required by an activity on a discrete basis over a time interval [t, t'], we state the following FOL axiom:- [Note: the "lost opportunity cost" due to a resource being disenabled is not considered in this chapter. However, a more detailed analysis is presented in Chapter 5.]

Axiom: $\forall a, r, c, c', t, t'$, *disenabled_res_cost(a, r, c, t)* $\wedge$ *discrete_mode (r, a)* $\wedge$ *[use_spec (r, a, t, t', q, rate, unit)* $\vee$ *consume_spec (r, a, t, t', q, rate, unit)]* $\wedge$ *status (s, a, disenabled, t')* $\wedge \neg$ *occursBet (disenabled (s, a), t, t')* $\supset$ *[disenabled_res_cost (a, r, c', t')* $\equiv$ *disenabled_res_cost_unit (a, r, q, v)* $\wedge$ $c' = c + v.q]$ (FOL 23)

For a reenabled state resource required by an activity on a continuous basis over a time interval [t, t'], we state the following FOL axiom:-

Axiom 25: $\forall a, r, c, c', t, t'$, *reenabled_res_cost(a, r, c, t)* $\wedge$ *continuous_mode (r, a)* $\wedge$ *[use_spec (r, a, t, t', q, rate, unit)* $\vee$ *consume_spec (r, a, t, t', q, rate, unit)]* $\wedge$ *status(s, a, reenabled, t')* $\wedge \neg$ *occursBet (reenabled (s, a), t, t')* $\supset$ *[reenabled_res_cost (a, r, c', t')* $\equiv$ *reenabled_res_cost_unit (a, r, q, v)* $\wedge$ $c' = c + v.(t' - t).rate]$ (FOL 24)

For a reenabled state resource required by an activity on a discrete basis over a time interval [t, t'], we state the following FOL axiom:-

Axiom: $\forall a, r, c, c', t, t'$, *reenabled_res_cost(a, r, c, t)* $\wedge$ *discrete_mode (r, a)* $\wedge$ *[use_spec (r, a, t, t', q, rate, unit)* $\vee$ *consume_spec (r, a, t, t', q, rate, unit)]* $\wedge$ *status(s, a, reenabled, t')* $\wedge \neg$ *occursBet (reenabled(s,a), t, t')* $\supset$ *[reenabled_res_cost (a, r, c', t')* $\equiv$ *reenabled_res_cost_unit (a, r, q, v)* $\wedge$ $c' = c + v.q]$ (FOL 25)

4.9.4 Aggregating Resource Costs for Multiple Intervals

A resource may be disenabled and reenabled several times before an activity is completed. Hence, there may be multiple intervals over which the state of a resource may be disenabled and reenabled. We must aggregate the costs for each of these intervals to compute the total cost for the disenabled and reenabled status of the states. To aggregate costs of a resource over several intervals of time, the predicates total_disenabled_res_cost and total_reenabled_res_cost are used to deduce the total disenabled resource cost and the total reenabled resource cost.

Axiom: The total disenabled resource cost of a resource at time t is equivalent to the sum of its disenabled resource cost for each of its disenabled time intervals that occurred prior to t. We state this axiom in FOL as follows:-

Axiom (schema): $\forall r, a, t, c', c_1, c_2..c_n, t_1, t_2,...t_n$ $total\_disenabled\_res\_cost(a, r, c', t) \equiv$ $[disenabled\_interval(a, r, t_1, t_2) \wedge disenabled\_interval(a, r, t_3, t_4) \wedge \ldots \wedge disenabled\_interval(a, r, t_{n-1}, t_n)] \wedge [disenabled\_res\_cost(a, r, c_1, t_2) \wedge disenabled\_res\_cost(a, r, c_2, t_4) \wedge \ldots \wedge disenabled\_res\_cost(a, r, c_n, t_n)] \wedge [t_1 < t_2 < \ldots < t_n \leq t] \wedge [c' = c_1 + c_2 + \ldots + c_n]$ (FOL 26)

Axiom: The total reenabled resource cost of a resource at time t is equivalent to the sum of its reenabled resource cost for each of its reenabled time intervals that occurred prior to t. We state this axiom in FOL as follows:-

Axiom (schema): $\forall r, a, t, c', c_1, c_2..c_n, t_1, t_2,...t_n, total\_reenabled\_res\_cost(a,r,c',t) \equiv$ $[reenabled\_interval(a, r, t_1, t_2) \wedge reenabled\_interval(a, r, t_3, t_4) \wedge \ldots \wedge reenabled\_interval(a, r, t_{n-1}, t_n)] \wedge [reenabled\_res\_cost(a, r, c_1, t_2) \wedge reenable\_res\_cost(a, r, c_2, t_4) \wedge \ldots \wedge reenabled\_res\_cost(a, r, c_n, t_n)] \wedge [t_1 < t_2 < \ldots < t_n \leq t] \wedge c' = c_1 + c_2 + \ldots + c_n$ (FOL 27)

However, a resource is committed and enabled only once to an activity before the activity is completed. Hence, the total_ committed_res_cost and the total_enabled_res_cost for an activity at time point t' is equivalent to the committed_res_cost at t' and the enabled_res_cost at t' respectively. Therefore, the total committed resource cost and the total enabled resource cost may be deduced from the following FOL axioms:-

Axiom: $\forall$ ***r, a, t, c, total_committed_res_cost (a, r, c, t)*** $\equiv$ ***committed_res_cost (a, r, c, t)*** (FOL 28)

Axiom: $\forall$ ***r, a, t, c, total_enabled_res_cost (a, r, c, t)*** $\equiv$ ***enabled_res_cost (a, r, c, t)*** (FOL 29)

4.10 Activity Costs

An activity may use or consume n different resources. The cost of an activity being dormant, executing, suspended and reExecuting at time t is computed by the aggregation of the total_committed_res_cost, the total_enabled_res_cost, the total_disenabled_res_cost, and the total_reenabled_res_cost at time t respectively for each of the n resources that is used or consumed by the activity. These computations are formalized with the following axioms:-

Axiom: The dormant_act_cost of an activity at time t is equivalent to the sum of the total_committed_res_cost for each of its resources. We state in FOL as follows:-

Axiom (schema): For each activity, a, we have an axiom of the form: $\forall a, c, c', r_1, r_2, \ldots r_n, t$, ***dormant_act_cost*** $(a, c', t) \equiv [$***total_committed_res_cost*** $(a, r_1, c_1, t) \wedge$ ***total_committed_res_cost*** $(a, r_2, c_2, t) \wedge$ ***total_committed_res_cost*** $(a, r_3, c_3, t) \wedge \ldots\ldots\ldots \wedge$ ***total_committed_res_cost*** $(a, r_n, c_n, t)] \wedge c' = c_1 + c_2 + \ldots + c_n$ (FOL 30)

Axiom: The execute_act_cost of an activity at time t is equivalent to the sum of the total_enabled_res_cost for each of its resources. We state in FOL as follows:-

Axiom (schema): For each activity, a, we have an axiom of the form: $\forall a, c, c', r_1, r_2, \ldots r_n, t$, $execute\_act\_cost(a, c', t) \equiv [total\_enabled\_res\_cost(a, r_1, c_1, t) \wedge total\_enabled\_res\_cost(a, r_2, c_2, t) \wedge total\_enabled\_res\_cost(a, r_3, c_3, t) \wedge \ldots \wedge total\_enabled\_res\_cost(a, r_n, c_n, t)] \wedge c' = c_1 + c_2 + \ldots + c_n$ (FOL 31)

Axiom: The suspend_act_cost of an activity at time t is equivalent to the sum of the total_disenabled_res_cost for each of its resources. We state in FOL as follows:-

Axiom (schema): For each activity, a, we have an axiom of the form: $\forall c, c', r_1, r_2, \ldots r_n, t$, $suspend\_act\_cost(a, c', t) \equiv [total\_disenabled\_res\_cost(a, r_1, c_1, t) \wedge total\_disenabled\_res\_cost(a, r_2, c_2, t) \wedge total\_disenabled\_res\_cost(a, r_3, c_3, t) \wedge \ldots \wedge total\_disenabled\_res\_cost(a, r_n, c_n, t)] \wedge c' = c_1 + c_2 + \ldots + c_n$ (FOL 32)

Axiom: The reExec_act_cost of an activity at time t is equivalent to the sum of the total_reenabled_res_cost for each of its resources. We state in FOL as follows:-

Axiom (schema): For each activity, a, we have an axiom of the form: $\forall c, c', r_1, r_2, \ldots r_n, t$, $reExec\_act\_cost(a, c', t) \equiv [total\_reenabled\_res\_cost(a, r_1, c_1, t) \wedge total\_reenabled\_res\_cost(a, r_2, c_2, t) \wedge total\_reenabled\_res\_cost(a, r_3, c_3, t) \wedge \ldots \wedge total\_enabled\_res\_cost(a, r_n, c_n, t)] \wedge c' = c_1 + c_2 + \ldots + c_n$ (FOL 33)

4.11 Cost Point of Activity: cpa(a, c, t)

The cost point of an activity, **cpa(a, c, t),** at time t may be obtained as the sum of the dormant, execute, suspended and reExecute activity costs for the activity at the time t. Hence, the computation for the cost point value, c, for the activity, a, at time point, t, may be axiomatized as follows:-

Axiom: The cost point of an activity at time t, denoted cpa(a, c, t) is equivalent to the sum of the dormant, execute, suspended and reExecute activity costs for the activity at the time t.

Stated in FOL, we have:-

$$\textit{Axiom: } \forall\, a, c_1, c_2, c_3, c_4, c', t, \; cpa\,(a, c', t) \equiv total\_dormant\_act\_cost(a, c_1, t) \wedge total\_execute\_act\_cost(a, c_2, t) \wedge total\_suspend\_act\_cost\,(a, c_3, t) \wedge total\_reExec\_act\_cost\,(a, c_4, t) \wedge [c' = c_1 + c_2 + c_3 + c_4] \qquad \text{(FOL 34)}$$

4.12 Cost Point of Resource for an Activity: cpr (a, c, t, r)

Axiom: The cost point of resource r for activity a at time t, denoted **cpr(a, c, t, r)**, is the sum of the total_committed_res_cost, total_enabled_res_cost, total_disenabled_res_cost and total_reenabled_res_cost at time point t.

Stated in FOL, we have:-

$$\textit{Axiom: } \forall\, a, c, c_1, c_2, c_3, c_4, t, \; cpr\,(a,c,t,r) \equiv total\_committed\_res\_cost\,(a,r,c_1,t) \wedge total\_enabled\_res\_cost\,(a, r, c_2, t) \wedge total\_disenabled\_res\_cost\,(a, r, c_3, t) \wedge total\_reenabled\_res\_cost\,(a, r, c_4, t) \wedge [c' = c_1 + c_2 + c_3 + c_4] \qquad \text{(FOL 35)}$$

4.13 Equivalence between cpa and cpr

The cost point of an activity at time t may also be achieved as the aggregation of all resource cost points of all resources used or consumed by the activity as was stated in Axiom (FOL 4).

Axiom: The cost point of an activity, cpa(a, c, t) is equivalent to the sum of the cost point of resource, cpr(a, c, t, r) for each of its n resources.

Stated in FOL, we have Axiom (schema) FOL 4 restated below:-

Axiom (schema) (FOL 4): For each activity, a, and $\forall c_1, c_2, c_3, ..., c_n, r_1, r_2, ... r_n, t, \exists c,$

$cpa(a, c, t) \equiv cpr(a, c_1, t, r_1) \wedge cpr(a, c_2, t, r_2) \wedge cpr(a, c_3, t, r_3) \wedge ... \wedge cpr(a, c_n, t, r_n) \wedge c = c_1 + c_2 + ... + c_n$

4.14 Cost Point of Cost Order

In ABC, activity costs are assigned to their cost object. Earlier, we rationalized an equivalence in the concept of cost object in ABC to cost order in our Cost Ontology for TOVE.

If several activities are performed towards a specific cost order, it is reasonable to assign the cost of all those activities to that cost order. Therefore, we state the following definition for cost point of cost order:-

> Definition: The *cost point of cost order* predicate **cpo (c, x, t)**, denotes c as the cost point of cost order x at time t, and is the aggregate cost of all activities for the cost order up to time t.
>
> We state cpo definition in FOL as follows:-

Axiom (schema): For each cost order, x, we have an axiom of the form: $\forall x, c', a_1, a_2, ..., a_n, c_1, c_2, ..., c_n, t_1, t_2, ..., t_n, t,\ cpo(c', x, t) \equiv [has\_cost\_order(x, a_1, t_1) \wedge cpa(a_1, c_1, t_1)] \wedge [has\_cost\_order(x, a_2, t_2) \wedge cpa(a_2, c_2, t_2)] \wedge .... \wedge [has\_cost\_order(x, a_n, t_n) \wedge cpa(a_n, c_n, t_n)] \wedge [t_1 < t_2 < .... < t_n \leq t] \wedge [c = c_1 + c_2 + c_3 + ... + c_n]$ **(FOL 36)**

4.15 Extending the Cost Ontology for ABM

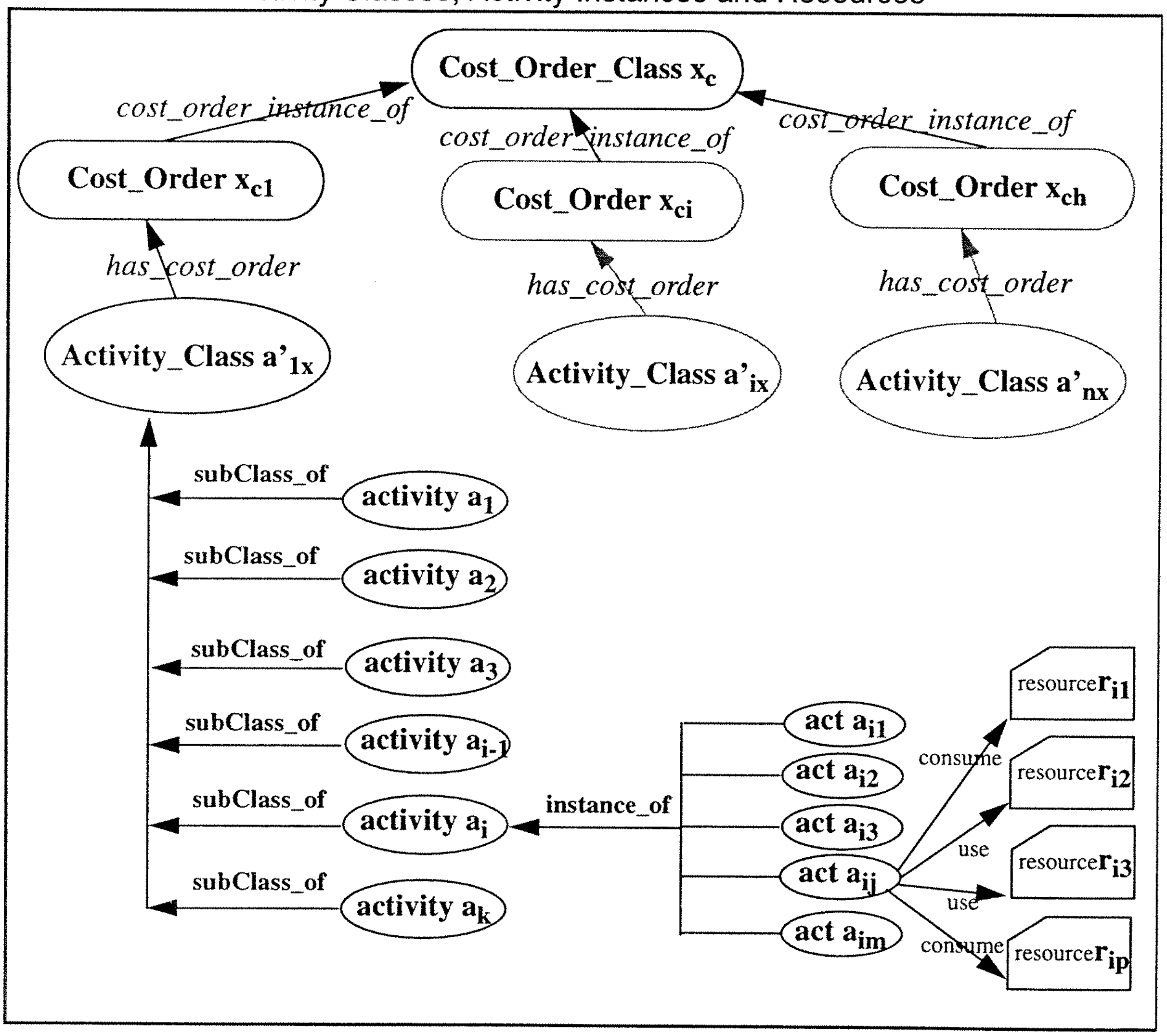
The competency questions of Sec. 4.2 gave guidance and direction to the Cost Ontology thus developed. The axioms developed so far enable us to compute and deduce costs for an instance of an activity, a, and an instance of an order, x. Cost computations pertaining to an instance of an activity, a, and an instance of an order, x, have been achieved through the set of axioms in the Cost Ontology that led to the formulation of cpa(a, c, t) and cpo(c, x, t) respectively. Hence, thus far, answers to some of the following common sense (competency) queries are obtainable:-

1. What is the instantaneous and cumulative cost of a resource used in an activity a at time t?
2. What is the instantaneous and cumulative cost of an activity a at time t?
3. What is the instantaneous and cumulative cost of an order o at time t?
4. What is the cost of resource at a point in time when it is being consumed or used at some continuous rate of consumption or usage?
5. What is the cost of the resource if it is required as some discrete or batch quantity by an activity?
6. What is the cost effect on a resource, given that the condition or state of the resource changes with time?
7. What cost effect does the state of a resource have upon the cost of an activity?
8. What cost effect does the state of a resource have upon the cost of a cost object?

However, computing and deducing costs for answers to queries 9, 10, 11, 12 and 13 of Sec.4.2 involves the aggregation of costs at various activity levels (viz., activity instance, subClass activity, and Class_Activity) and order levels (viz., cost order instance, and Cost_Order_Class). Finding answers to such queries may be essential to achieve strategic cost management for multi-national, multi-subsidiary corporations. Hence, though competency queries 9 - 13 of Sec.4.2 are not meant to be totally exhaustive and mutually exclusive, they serve as examples that strongly motivate the need to extend the Cost Ontology to provide the aggregation of costs through the various levels of activity and cost_order representations shown in Figure 9.

Fig. 9 illustrates that activity classes, a'_1, a'_2, a'_3,.... a'_n has_cost_order_class, x_c, up to time t. Activities a_1, a_2, a_3,......, a_k are subClass activities of Class_Activity of a'_{1x}; and a_{i1}, a_{i2}, a_{i3}, ..., a_{im} are instances_of a_i. Each instance, a_{ij}, of activity, a_i, uses or consumes resources r_{i1}, r_{i2}, r_{i3},......,r_{ip}.

FIGURE 9 Relationships up to time t amongst Cost Order Classes, Activity Classes, Activity Instances and Resources



4.15.1 Computing Cost Point of subClass Activities

From Axiom (FOL 33), for each resource, r_{ip}, required by a_{ij}, the resource cost point, cpr, is the sum of the total committed, enabled, disenabled and reenabled resource cost. Therefore,

$$\textit{Axiom:} \forall\, a, r_{ip}, c', c_1, c_2, c_3, c_4, t,\ cpr(a, c', t, r_{ip}) \equiv total\_committed\_res\_cost\,(a, r_{ip}, c_1, t) \wedge total\_enabled\_res\_cost(a, r_{ip}, c_2, t) \wedge total\_disenabled\_res\_cost\,(a, r_{ip}, c_3, t) \wedge total\_reenabled\_res\_cost(a, r_{ip}, c_4, t) \wedge [c' = c_1 + c_2 + c_3 + c_4] \quad \text{(FOL 37)}$$

From Axiom Schema (FOL 2), given that activity instance, a_{ij}, requires p different resources, the cost point of activity instance, a_{ij}, is the aggregation of cost point of each of the p resources.

$$\textit{Axiom (schema): For each activity instance, } a_{ij}\textit{, the axiom is of the form:- } \forall\, c, c_1, c_2, c_3, \ldots, c_p, r_{i1}, r_{i2}, \ldots r_{ip}, t,\ cpa\,(a_{ij}, c, t) \equiv cpr\,(a_{ij}, c_1, t, r_{i1}) \wedge cpr\,(a_{ij}, c_2, t, r_{i2}) \wedge cpr\,(a_{ij}, c_3, t, r_{i3}) \wedge \ldots \wedge cpr\,(a_{ij}, c_p, t, r_{ip}) \wedge [c = c_1 + c_2 + \ldots + c_p] \quad \text{(FOL 38)}$$

The cost point of a subClass activity, a_i, is the aggregation of the cost point of each activity instance, a_{ij}. This computation may be axiomatized with the use of the distinguishing predicate, **cpa_subClass**, as follows:-

$$\textit{Axiom (schema) : For each subClass activity, } a_i\textit{, the axiom is of the form:- } \forall\, c_i, t, a_{i1}, a_{i2}, \ldots, a_{im}, c_{i1}, c_{i2}, \ldots, c_{im},\ [a_{ij} \neq a_{ik}, j \neq k],\ instance\_of\,(a_i, a_{i1}) \wedge instance\_of\,(a_i, a_{i2}) \wedge instance\_of\,(a_i, a_{i3}) \wedge \ldots \wedge instance\_of\,(a_i, a_{im}) \supset cpa\_subClass\,(c_i, a_i, t) \equiv cpa\,(a_{i1}, c_{i1}, t) \wedge cpa(a_{i2}, c_{i2}, t) \wedge cpa\,(a_{i3}, c_{i3}, t) \wedge \ldots \wedge cpa\,(a_{im}, c_{im}, t) \wedge [c_i = c_{i1} + c_{i2} + c_{i3} + \ldots + c_{im}] \quad \text{(FOL 39)}$$

4.15.2 Computing Cost Points of Class Activities for Cost Orders

The cost point of Activity Class, a'_{ix}, which is the ith class activity for cost order, x_{ci}, is the aggregation of cost point of each subClass activity, a_i. This procedure is formalized as follows using the distinguishing predicate, **cpa_Class,** to indicate the cost point of an activity class:-

Axiom (schema): For each Class activity, a'_{ix}, the axiom is of the form:- $\forall c'_i, t, a_1, a_2, \ldots, a_k, c_1, c_2, \ldots, c_k$ $subClass\_of(a'_{ix}, a_1) \wedge subClass\_of(a'_{ix}, a_2) \wedge \ldots \wedge subClass\_of(a'_{ix}, a_k) \supset cpa\_Class(c'_1, a'_{ix}, t) \equiv cpa(a_1, c_1, t) \wedge cpa(a_2, c_2, t) \wedge cpa(a_3, c_3, t) \wedge \ldots \wedge cpa(a_k, c_k, t) \wedge [c'_i = c_1 + c_2 + c_3 + \ldots + c_k]$ **(FOL 40)**

4.15.3 Computing Cost Points of Cost Orders

Given that the process plan for an order, x, has specified class activities, $a'_{1x}, a'_{2x}, \ldots, a'_{nx}$, the cost point of x is the aggregation of the cost point of each class activity specified in the process plan of x. Thus, applying Axiom 41, the cost point of an order (or an instance of an order) as:-

Axiom (schema) 46: For each cost order, x, we have an axiom of the form:- $\forall c'_x, a'_{1x}, a'_{2x}, \ldots, a'_{nx}, c'_{1x}, c'_{2x}, \ldots, c'_{nx}, t_1, t_2, \ldots, t_n, t, cpo(c'_x, x, t) \equiv [has\_cost\_order(x, a'_{1x}, t_1) \wedge cpa(a'_{1x}, c'_{1x}, t_1)] \wedge [has\_cost\_order(x, a'_{2x}, t_2) \wedge cpa(a'_{2x}, c'_{2x}, t_2)] \wedge \ldots \wedge [has\_cost\_order(x, a'_{nx}, t_n) \wedge cpa(a'_{nx}, c'_{nx}, t_n)] \wedge [t_1 < t_2 < \ldots < t_n \leq t] \wedge [c'_x = c'_{1x} + c'_{2x} + c'_{3x} + \ldots + c'_{nx}]$ **(FOL 41)**

4.15.4 Computing Cost Points of Cost Order Classes

As illustrated in Fig. 22, cost order class, x_c, may be comprised of h number of cost order instances, x_{c1}, x_{c2}, ... x_{ci}, ...x_{ch}. Hence, each instance, x_{ci}, is a cost_order_instance_of cost order class, x_c. The costing of cost order class, x_c, is the aggregation of the cost order instances, x_{ci}'s. To formalize the aggregation of the cost order instances for the cost point of x_c, we use the distinguishing predicate, **cpo_Class**, to indicate the cost point of cost order class, x_c as follows:-

Axiom (schema): For each cost order class, x_c, we have an axiom of the form:- $\forall c_{xc}, t, x_{c1}, x_{c2}, \ldots\ldots, x_{ch}, c_{x1}, c_{x2}, \ldots\ldots, c_{xh}$, *cost_order_instance_of* $(x_c, x_{c1}) \wedge$ *cost_order_instance_of* $(x_c, x_{c2}) \wedge \ldots\ldots\ldots\ldots \wedge$ *cost_order_instance_of* $(x_c, x_{ch}) \supset$

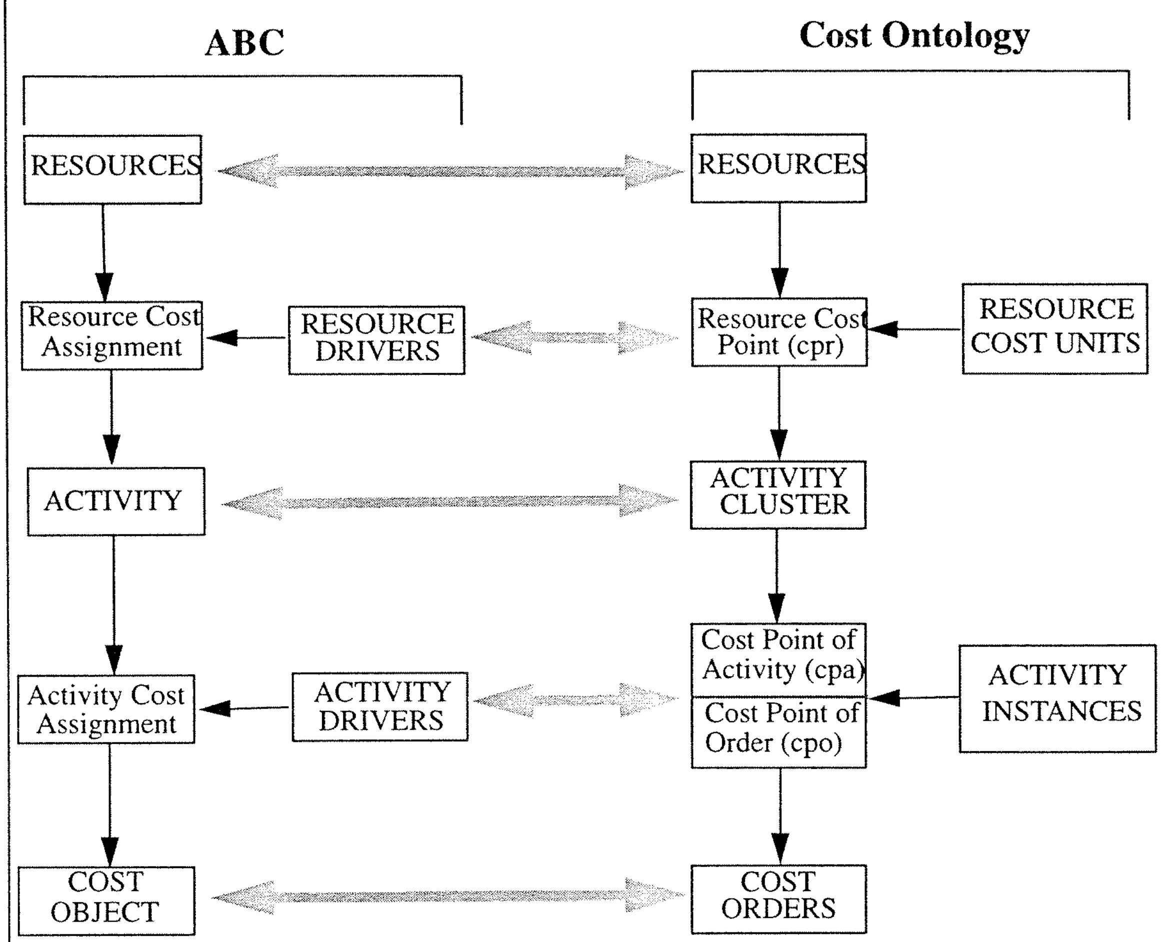
$$cpo\_Class\ (x_c, c_{xc}, t) \equiv cpo\ (c_{x1}, x_{c1}, t) \wedge cpo\ (c_{x2}, x_{c2}, t) \wedge cpo\ (c_{x3}, x_{c3}, t) \wedge \ldots\ldots\ldots\ldots\ldots\ldots \wedge cpo\ (c_{xh}, x_{ch}, t) \wedge [c_{xc} = c_{x1} + c_{x2} + c_{x3} + \ldots\ldots + c_{xh}] \quad \text{(FOL 42)}$$

4.16 Mapping the ABC Principle to the Cost Ontology

Resources are considered as the necessary requirements to accomplish or to perform an activity [Fadel & Fox 94]. In that sense, the property of the resource is dependent on the activity to be performed. Some examples of resources are machines, computers, materials, tools, humans, floor space, electricity, etc. However, *from a cost perspective, resources are the sources of cost and are viewed as economic elements directed to the performance of activities.*

The *resource drivers* are "the links between the resources and activities. They take a cost from the general ledger and assign it to the activities"[Turney 92]. As many resources may be consumed or used by an activity, an activity may have several resource drivers. Looking for resource drivers, which are transaction-related "cost drivers", forms the first stage in cost management that helps management discover what contributes to costs [Stoffel 92]. *As illustrated in figure 23, the Cost Ontology enables the mapping of resource drivers to resource cost units; whereas, the resource cost assignment of ABC is achieved through the cost micro-theory for the cost point of resource (cpr) of an activity.*

FIGURE 10 Mapping the Conceptualization* of ABC with the Cost Ontology



***Based on [Turney 92]&[Turney 92]**

In the ABC context, activity is considered "a combination of people, technology, raw materials, methods, and environment that produces a given product or service" [Brimson 91]. The development of Cost Ontology centers around the more precise and complete representation of the activity cluster.

In ABC, the reason for performing an activity is considered a cost object [Turney 92, ABC Glossary for CAM-I, Arlington, Texas]. A cost object is the reason why work is performed by an enterprise. Products and customers are reasons for performing activities. Cost objects include products, services, customers, projects and contracts.The cost object is the terminal point to which cost is traced. Consequently, the cost traced to each cost object will reflect the cost of the activities used by that cost object. *The concept of orders in TOVE is similar to the concept of cost object in ABC. From the perspective of costing and the Cost Ontology, the class of order in TOVE forms the cost order for activities for ABC purposes.*

In the ABC concept, each activity is traced to the cost object via an activity driver. *An activity driver is a measure of the consumption or usage of an activity by a cost object.* For example, the number of hours devoted by the design engineers to design a product may be considered as the activity driver for the engineering design activity.

ABC assigns the cost of activities to cost objects based on activity drivers that accurately measure consumption or usage of the activity. Cost objects are costed accurately when activity drivers measure the use of activities directly or correlate closely with their use. Hence, for the purpose of activity cost assignment to a cost object, the activity driver is used to assign resources from the activities to the cost objects. Identifying the most appropriate activity driver, which is also considered a transaction-related "cost driver", for an activity consumed by a cost object, forms the second stage of cost allocation in ABC that helps management discover what contributes to cost. Our representation of an activity cluster, together with the "use or consume" specifications of resources and the computations for the resource cost point of an activity, enable us to compute the cost point of an activity (cpa) and the cost point of a cost order (cpo) through the precise and complete representation of an activity instance. Hence, *an activity instance in our Cost Ontology serves the purpose of assigning activity costs to an instance of a cost order just as an activity driver in ABC serves the purpose of activity cost assignment to a cost object.*

4.16.1 Micro-theory of ABC

A micro-theory is a formal model based upon an ontology to do a specific task. The task at hand is the activity-based cost of a cost object or cost order. Figure 23 indicates the cpr, cpa, and cpo founded upon the Cost Ontology for TOVE does accomplish the task at hand.

In short, a Micro-theory of ABC includes **cpr, cpa** and **cpo** that are founded upon the Cost Ontology in TOVE

4.17 Summary

In summary, the conceptualization of ABC provides a framework to providing cost and operational information about the work carried on by the enterprise to be modelled. From the enterprise modelling perspective, it advantageously encompasses the following building blocks of the enterprise for their indicated purposes:-

- Resources for Resource Management,
- Resource Drivers for Cost Management and Business Process Re-design,
- Activities for Activity Management and Business Process Engineering,
- Activity Drivers for Cost Management and Business Process Re-design,
- Cost Objects for Strategic Management of the enterprise.

Therefore, the ABC framework of cost management for enterprise modelling points directly to profit opportunities by revealing the links from resource consumption to activities via resource drivers, and from activities to cost objects via activity drivers [Cooper & Kaplan 91].

In Fig. 10, the mapping of the ABC conceptualization to the Cost Ontology for TOVE and a Micro-theory of ABC is evidenced by the following:-

1. Resource Drivers of ABC to the committed_res_cost_unit, enabled_res_cost_unit, disenabled_res_cost_unit, reenabled_res_cost_unit of a resource;
2. Cost Objects of ABC to the Cost Orders in TOVE;
3. Activity Costs of ABC to the temporal and traceable dormant_act_cost, execute_act_cost, suspend_act_cost, reExec_act_cost;
4. the assignment of activity costs to instances of cost orders through the developed Micro-theory of ABC involving cost point of a resource (cpr), cost point of activity (cpa), and cost point of order (cpo);

5. the aggregation of activity costs for a Class Activity and Cost Order Class through the extension of the Cost Ontology to include distinguishing predicates, cpa_subClass, cpa_Class, and cpo_Class, for the activity cost computations of cost point of subClass activity, cost point of Class Activity and cost point of Cost Order Class respectively.

4.18 Closure Axioms

ABC assigns costs to activities based on their use of resources. The logical formulation of ABC must be premised on the existence and knowledge of the resource cost units. Closure for the activity cost management domain must be ensured by the complete knowledge (completeness) of time, activities, status of states, resources and their respective quantities required for activities together with the resource cost units for each of the resources. It is only with this complete information, is it possible to achieve activity based cost management.

Depending on the status value of the activity, each represented activity of a particular enterprise may require various resources. Hence, from a cost perspective, the following schema of closure axioms, expressed in first order logic, are relevant to the particular enterprise being modelled. This formalization ensures that the resource cost units must be inputs as part of the data model that links each resource with each activity for the computations and deductions of activity costs towards the determination of resource cost points (cpr) of an activity, the cost point of an activity (cpa), and the cost point of an order (cpo).

Closure Axiom (schema) 1: The axiom for the committed resource cost unit is of the form:-∀a, r, q, v, committed_res_cost_unit(a, r, q, v) ≡ (a = activity_1 ∧ r = resource_11 ∧ q = qty_11 ∧ v = value_11) ∨ (a = activity_1 ∧ r = resource_12 ∧ q = qty_12 ∧ v = value_12) ∨ (a = activity_1 ∧ r = resource_13 ∧ q = qty_13 ∧ v = value_13) ∨...........∨ (a = activity_1 ∧ r = resource_1p ∧ q= qty_1p ∧ v = value_1p) ∨..........................∨ (a = activity_n ∧ r = resource_n1 ∧ q = qty_n1

$\wedge$ *v = value_n1)* $\vee$ *(a = activity_n* $\wedge$ *r = resource_n2* $\wedge$ *q = qty_n2* $\wedge$ *v = value_n2)* $\vee$ *(a = activity_n* $\wedge$ *r = resource_n3* $\wedge$ *q = qty_n3* $\wedge$ *v = value_n3)* $\vee$...............$\vee$ *(a = activity_n* $\wedge$ *r = resource_ny* $\wedge$ *q = qty_ny* $\wedge$ *v = value_ny)*

where n enterprise activities modelled, activity_1 requires p different resources and has committed resource cost unit values value_11, value_12, value_13,... value_1p, etc.,respectively and activity_n requires y different resources and has y committed resource cost unit values value_n1, value_n2, value_n3, ... value_ny.

Closure Axiom (schema) 2: The axiom for the enabled resource cost unit is of the form:- $\forall$ ***a, r, q, v, enabled_res_cost_unit(a,r,q,v)*** $\equiv$ ***(a = activity_1*** $\wedge$ ***r = resource_11*** $\wedge$ ***q = qty_11*** $\wedge$ ***v = value_11')*** $\vee$ ***(a = activity_1*** $\wedge$ ***r = resource_12*** $\wedge$ ***q = qty_12*** $\wedge$ ***v = value_12')*** $\vee$ ***(a = activity_1*** $\wedge$ ***r = resource_13*** $\wedge$ ***q =qty_13*** $\wedge$ ***v = value_13')*** $\vee$............$\vee$ ***(a = activity_1*** $\wedge$ ***r = resource_1p*** $\wedge$ ***q = qty_1p*** $\wedge$ ***v = value_1p')*** $\vee$.................$\vee$ ***(a = activity_n*** $\wedge$ ***r = resource_n1*** $\wedge$ ***q = qty_n1*** $\wedge$ ***v = value_n1')*** $\vee$ ***(a = activity_n*** $\wedge$ ***r = resource_n2*** $\wedge$ ***q=qty_n2*** $\wedge$ ***v = value_n2')*** $\vee$***(a =activity_n*** $\wedge$ ***r = resource_n3*** $\wedge$ ***q = qty_n3***$\wedge$***v = value_n3')*** $\vee$..............$\vee$ ***(a = activity_n*** $\wedge$ ***r = resource_ny*** $\wedge$ ***q = qty_ny*** $\wedge$ ***v = value_ny'),***

where n enterprise activities modelled, activity_1 requires p different resources and has p exececute resource cost unit values value_11', value_12', value_13',... value_1p', etc., respectively and activity_n requires y different resources and has y resource cost unit values value_n1', value_n2', value_n3', ... value_ny'.

Similarly, we have Closure Axioms 3 and 4 in the following forms:-

Closure Axiom (schema) 3: The axiom is of the form:- $\forall$ ***a, r, q, v, disenabled_res_cost_unit(a,r,q,v)*** $\equiv$ ***(a = activity_1*** $\wedge$ ***r = resource_11*** $\wedge$ ***q = qty_11*** $\wedge$ ***v= value_11")*** $\vee$ ***(a = activity_1*** $\wedge$ ***r = resource_12*** $\wedge$ ***q = qty_12*** $\wedge$ ***v = value_12")*** $\vee$ ***(a = activity_1*** $\wedge$ ***r = resource_13*** $\wedge$ ***q = qty_13***$\wedge$ ***v = value_13")*** $\vee$............$\vee$ ***(a = activity_1*** $\wedge$ ***r = resource_1p*** $\wedge$ ***q = qty_np*** $\wedge$ ***v = value_1p")***

$\vee$............................$\vee$ *(a = activity_n $\wedge$ r = resource_n1 $\wedge$ q = qty_n1 $\wedge$ v = value_n1") $\vee$ (a = activity_n $\wedge$ r = resource_n2 $\wedge$ q=qty_n2 $\wedge$ v= value_n2") $\vee$(a = activity_n $\wedge$ r = resource_n3 $\wedge$ q = qty_n3 $\wedge$ v = value_n3")* $\vee$..............$\vee$ *(a = activity_n $\wedge$ r = resource_ny $\wedge$ q = qty_ny $\wedge$ v = value_ny")*

Closure Axiom (schema) 4: The axiom is of the form:- $\forall$***a, r, q, v, reenabled_res_cost_unit (a,r,q,v)*** $\equiv$ ***(a = activity_1 $\wedge$ r = resource_11 $\wedge$ q = qty_11 $\wedge$ v= value_11"') $\vee$ (a = activity_1 $\wedge$ r = resource_12 $\wedge$ q = qty_12 $\wedge$ v =value_12"') $\vee$ (a = activity_1 $\wedge$ r = resource_13 $\wedge$ q = qty_13$\wedge$ v = value_13"')*** $\vee$..............$\vee$ ***(a = activity_1 $\wedge$ r = resource_1p $\wedge$ q = qty_np $\wedge$ v = value_1p"')***$\vee$...................$\vee$ ***(a = activity_n $\wedge$ r = resource_n1 $\wedge$ q = qty_n1 $\wedge$ v = value_n1"') $\vee$ (a = activity_n $\wedge$ r = resource_n2 $\wedge$ q = qty_n2 $\wedge$ v = value_n2"') $\vee$(a =activity_n $\wedge$ r = resource_n3 $\wedge$ q = qty_n3 $\wedge$ v = value_n3"')*** $\vee$..............$\vee$ ***(a = activity_n $\wedge$ r = resource_ny $\wedge$ q = qty_ny $\wedge$ v = value_ny"').***

4.19 Concluding Remarks

This chapter has described a Cost Ontology for TOVE. A Micro-theory of ABC has been established for enterprise modelling that spans the knowledge representation of activity, status of activity, time, causality, and resources. The axioms and the forms of axioms presented in first order logic (FOL) have been specifically developed to answer competency questions that must be answered for the implementation of ABM.

The mapping of the ABC Principle to the Cost Ontology establishes a Micro-theory of ABC. This makes it possible to consistently and unambiguously reason, deduce and compute activity-based costs for the activities and cost objects of any enterprise. [This has been achieved through the coding of the axioms in Quintus Prolog to facilitate the querying of the ROCK database for the TOVE factory manufacturing clip-lamps. The implemented axioms are used in the case studies with deHavilland and BHP companies as shown in chapters 7 and 8 respectively].

The computations of activity based costs for any cost object (cost order) is possible if resource cost units are known or given for the enterprise modelled. Closure axioms have been developed to maintain closure for the ABM domain by ensuring that complete knowledge (completeness) of time, activities, status of states, resources and their respective quantities required for activities together with the resource cost units for each resource is known. It is only with this complete information, is it possible to achieve the desired ABM as founded upon the competency queries that drove the Cost Ontology design.

The Cost Ontology and Micro-theory of ABC make it possible to reason, deduce and compute activity-based costs for the operations of an enterprise, independent of the industrial sector - manufacturing or service - to which an enterprise may belong to. This would not only make possible the effective management of resources and activities towards an enterprise satisfying its clients, but would also provide an evaluation costing tool for business process engineering (BPE) or business process re-engineering (BPR). Hence, the Cost Ontology for TOVE and Micro-theory of ABC form contributions to ABM and strategic cost management.

CHAPTER 5 Theory of Resource Cost Units

5.1 Introduction

The resource cost unit of a resource is the cost of a unit of the resource in the state that it exists in the real world at some time point. The commit_resource_cost_unit, the enabled_resource_cost_unit, the disenabled_resource_cost_unit and the re-enabled_resoure_cost_unit are respectively associated with the commit, enabled, disenabled and re-enabled states associated with a resource. The micro-theory of costs for the computations of activity-based costs assumes that resource cost units of a resource are known or given for the enterprise modelled. This chapter defines how resource cost units are computed.

Initially, a resource may come into existence in one of two ways - a resource is made available after the company has gone through the planning and production activities to produce the resource; or the resource is made available after the company has gone through some planning and purchase activities to acquire the resource from some external supplier. In other words, there must have been a set of activities completed prior to a resource being there. This brings forth the concept of the trace of a resource.

By definition, the trace of a resource consists of an activity set (termed commit envelope) that is composed of activities that must be completed prior to a resource being there, such that the last completed activity of this set produces the resource.

From an activity modelling perspective, the term commit envelope denotes an aggregate activity class that is composed of class activities *a1, a2, a3, ..., an* that form the trace of activities to a resource class r.

From an ABC perspective, it is reasonable that all activities that trace or mark a path towards the eventual production of a resource must be considered to contribute towards the cost of the resource. Further, the trace of the resource may include activities that contribute an insignificant cost towards the resource itself. On the other hand, since the cost of an activity depends on the cost of resources required by an activity, there may be a resource required by an activity whose cost contribution is insignificant towards the costing of the activity itself. Hence, intuitively, questions about significant activities and resources arise in the context of tracing activities to a resource for the purpose of costing a resource.

Therefore, the following issues have *motivated* this research to delve into the make-up or composition of resource cost units:-

1. How does one build a formal model of the trace of activities to a resource?
2. What is the extent of this trace of activities, i.e., where does the trace begin and where does it end?
3. In determining the cost of each activity in this trace, how are overhead costs accurately and consistently included keeping in mind the following:-

   "Advances in the technology of manufacturing planning, control and operation have made conventional cost accounting practices not only obsolete but dangerous. As a portion of total factory cost, direct labour costs have declined dramatically due to improved quality materials, few interruptions, automation and robotics. Overhead costs, conversely, have increased greatly and the fixed (overhead) components now dominate". [Plossl 90]

"Why, then, is the traditional approach to product costing (direct materials, direct labour and factory overhead via a burden rate) inappropriate in today's manufacturing environment? There are several reasons, (all related to the factors of change outlined above[1]):

- The factory floor is no longer a suitable boundary for the overhead which should be accounted for in product costs. The boundary should be extended to all those support overhead departments which account for more and more of total cost and carry out activities related to products, including design, sales order processing, through to delivery.
- Direct labour will not be the cost driver for many direct factory overheads and even less so for support overheads. Yet it is still the most common means of burdening product costs with overheads (see CAM-I Survey of Management Accounting). Instead, cost drivers are needed to attribute costs which are related to the activities performed. Product costs can then be built up by considering the activities required to manufacture a product (and the non-value added activities, which should be minimized)." [Jeans & Morrow 89]

According to Gary Cokins [96]: "In effect, traditional general ledger cost accounting systems act like thick *cloud covers*. The clouds prevent any observation, and eventual understanding, of the locations and rates at which the enterprise uses resources to enable the creation of value or to actually create value for customers."

In the review by Ruhl [97], 'cloud covers' in Cokin's simile stands for arbitrary overhead allocations, which fail to accurately reflect how different products, customers, and processes actually consume resources.

1 Some factors of change outlined in the same paper are: (a) decline of direct labour as a proportion of total cost; (b) increased indirect overhead (production planning, engineering, purchasing, etc.); (c) increase in capital investment, especially in technology, on the factory floor and in support departments.

4. Traditional accounting descriptions of indirect costs include terms such as "overhead" and "burden", both of which convey negative connotations. The term "overhead" implies that it is not related to or is disconnected from the real work of an enterprise, real work being the performance of activities towards achieving enterprise goals. The term "burden" implies some misfortune or handicap has befallen those unfortunate enough to bear it. Moreover, a distinction between direct cost and indirect costs seems meaningless to us through our deployment of the ABC principle within TOVE's core representation of an enterprise model based on activities, states and resources. For one thing, the ABC deployment in TOVE has taught us that much of "indirect costs" are directed to activities, and via the activities, direct to cost objects, i.e., products and customers. Further, the concept of a resource as presented in the Resource Ontology for TOVE [Fadel 94], presents an opportunity to cover conventional notions of overheads. After all, a resource in TOVE may represent a person, a computer, information, and other factors such as authorization and policy that enable an activity to be performed regardless of traditional distinctions or type of organization. Hence, in our quest for resource cost units, we are motivated to explore a resource taxonomy built upon the Resource Ontology for TOVE to include notions of overhead entities, and to build a theory that rids us of the negative connotations brought about by "indirect costs" by directing all costs to activities in TOVE, and then via activities to cost objects based upon the ABC principle.

5. What micro-theory of ABC enables one to accurately and consistently compute the resource cost units of a resource?

6. Should only significant activities and significant resources be considered towards our investigation of resource cost units of a resource? If so, how may we define a significant activity and a significant resource?

The above issues have motivated the development of the Theory of Resource Cost Units. In particular, the issues of activity trace to a resource and the extent of the trace have contributed to the formalization of the Principle of Resource Probe. The issue of costing activities together with appropriate "overhead" accountability to the activities performed, have contributed to the Formalization of Overheads and Activity Envelopes for a resource.

This chapter is about what contributes to the make up of resource cost units. The above motivating issues are explored within the Theory of Resource Cost Units that includes:

- Resource and Activity Probing,
- the Commit Envelope of an Internal Resource,
- the Reenable Envelope of a Resource,
- the Enveloped Activity Based Enterprise Model (EABEM),
- a Formalization of Overheads,
- Axioms to deduce the Resource Cost Units of a resource.

5.2 Assumptions

Since our domain of interest are costs to an enterprise based upon activities, states and resources of an enterprise, the following assumptions are made:-

Assumption 1: Knowledge of enterprise resources and activities that fall within the jurisdiction of the organization are known. In other words, the enterprise would have the knowledge of activities that they consider to be their activities to meet company goals along with the corresponding resources required by those activities.

- **Activity-known:** This is the most basic term in the Activity Ontology. It is similar to the term in the Resource Ontology [Fadel et al.94] defined **rknown(r)** that specifies knowledge of a resource as opposed to its physical existence. **act_known(a)** specifies *knowledge of an activity* regarding its past, present or future execution. The importance of its definition lies in the fact that the ability to reason about an activity and about its activity costs depends on it being known.

Axiom: $(\forall r), rknown(r)$ (FOL 43)

$$Axiom:\ (\forall a),\ act\_known(a) \qquad \text{(FOL 44)}$$

- **internal_res(r):** It is logical to assume that an enterprise has knowledge of its own internal resources. For purposes of ABC, an enterprise must have knowledge of its internal resources.

$$Axiom:\ (\forall r),\ internal\_res(r) \supset rknown(r) \qquad \text{(FOL 45)}$$

- **internal_act(a):** It is logical to assume that an enterprise has knowledge of its own internal activities. For our purposes of ABC, an enterprise must have knowledge of its internal activities.

$$Axiom:\ (\forall a),\ internal\_act(a) \supset act\_known(a) \qquad \text{(FOL 46)}$$

- **external_act(a):** It is logical to assume that an enterprise does not have knowledge of activities considered external to the organization.

$$Axiom:\ (\forall a),\ external\_act(a) \supset \neg\ act\_known(a) \qquad \text{(FOL 47)}$$

Assumption 1 may be stated and axiomatized as follows:-

All internal activities and their required resources must be known to the enterprise.

$$Axiom:\ \forall(a, r),\ internal\_act(a) \wedge requires\ (a, r) \supset act\_known(a) \wedge rknown(r) \qquad \text{(FOL 48)}$$

<u>*Assumption 2:*</u> Resources required by activities may be produced by the company's internal activities, or may be purchased or acquired directly from some external source.

This assumptions entails defining the following terms:-

- *An internal resource is one that is produced by an internal activity.*

$$Axiom:\ \forall\ r,\ internal\_res(r) \equiv (\exists\ a),\ [internal\_act(a) \wedge produces(a, r)] \qquad \text{(FOL 49)}$$

- *An external resource is one for which there exists no internal activity that produces it.*

$$Axiom:\ \forall\ r,\ external\_res(r) \equiv \neg\ internal\_res(r) \equiv \neg\ (\exists\ a),\ [internal\_act(a) \wedge produces(a, r)] \qquad \text{(FOL 50)}$$

Assumption 2 may be restated and axiomatized as follows:-

Each resource required by a known activity may only be an internal or an external resource.

$$\textit{Axiom: } (\forall\, a, r),\ act\_known\,(a) \wedge requires(a, r) \supset internal\_res(r) \vee external\_res(r) \quad \text{(FOL 51)}$$

Assumption 3: For all internal and external resources, there exists a unit cost c for the resource r. The unit cost of an internal resource is an activity-based cost, whereas the unit cost of an external resource is a nonactivity-based cost

To formalize Assumption 3 from a cost perspective, we focus upon a class of fluents termed cost_fluents. In general, *the term fluents refer to a class of atomic entities in dynamic environments.* Fluents repeatedly *change* their value as events occur and are functions of the situation or time [Schwalb et al. 95]. In general, *holds (f, s)* denotes that fluent *f* holds true in situation *s*.

changes (a, f): Changes in a fluent *f* from situation *s1* to situation *s2* are brought about by some action *a*.

The predicate changes(a, f) is axiomatized as follows:-

$$\textit{Axiom: } (\forall\, a, s1, s2, f),\ changes(a, f) \equiv Do(a, s1, s2) \wedge holds(f, s1) \supset \neg holds(f, s2) \quad \text{(FOL 52)}$$

This research recognizes that changes in costs from situation to situation may be brought about through some action (activity) or through some factor other than some action. Hence, we formally distinguish between activity cost fluents and nonactivity cost fluents.

- **activity_cost(f):** We define activity cost fluents as a cost fluent for which there exists some activity that brings about its change.

$$\textit{Axiom: } (\forall\, f),\ activity\_cost(f) \equiv (\exists\, a),\ changes(a, f) \wedge cost\_fluent(f) \quad \text{(FOL 53)}$$

- **nonactivity_cost(f)**: We define nonactivity cost fluents as a cost fluent for which there exist no activity that brings about its change.

Axiom: $(\forall f)$, ***nonactivity_cost (f)*** $\equiv \neg (\exists a)$, ***changes(a, f)*** $\wedge$ ***cost_fluent(f)*** (FOL 54)

unit_cost_int(c, r): is defined as a class of activity cost fluent, where *c* denotes cost of 1 unit of an internal resource *r*.

Axiom: $(\forall c, r)$, ***activity_cost (unit_cost_int (c, r))*** (FOL 55)

unit_cost_ext(c, r): is defined as a class of nonactivity cost fluent where *c* denotes the cost of 1 unit of an external resource *r*.

Axiom: $(\forall c, r)$, ***nonactivity_cost (unit_cost_ext (c, r))*** (FOL 56)

To formalize Assumption 3 for internal and external resources, we state in Axiom 57 that for all internal resources, there exists a unit cost of the resource that is an activity-based cost; whereas, Axiom 58 states that for all external resources, there exists a unit cost of the resource that is a nonactivity-based cost.

Axiom: $(\forall r)$, ***internal_res(r)*** $\supset (\exists c, s)$, ***holds(unit_cost_int(c, r), s)*** $\wedge$
activity_cost(unit_cost_int(c, r)) (FOL 57)

Axiom: $(\forall r)$, ***external_res(r)*** $\supset (\exists c, s)$, ***holds(unit_cost_ext(c, r), s)*** $\wedge$
nonactivity_cost(unit_cost_ext(c,r)) (FOL 58)

Since a resource known to an enterprise may be either an internal resource or an external resource, Assumption 3 may be alternately stated that all known resources to an enterprise must have a unit cost.

Axiom: $(\forall r)$, ***rknown(r)*** $\supset (\exists c, s)$, ***[holds(unit_cost_int(c, r), s)]*** $\vee$
[holds(unit_cost_ext (c, r), s)] (FOL 59)

<u>*Assumption 4:*</u> A resource cannot be required and be produced by the same activity.

Axiom: $(\forall a, r)$, ***requires(a, r)*** $\supset \neg$ ***produces(a,r)*** (FOL 60)

We also introduce two distinct classes of activities that will be used later in this chapter.

> **frontier_act(a):** A frontier activity is a known activity that has all of its enabling states require external resources only.

Axiom: $\forall\ (a, r),\ requires(a, r) \wedge external\_res(r) \equiv frontier\_act(a)$ (FOL 61)

> **internal_act(a):** An internal activity is one that is not a frontier activity, or stated differently, an internal activity is one that requires at least on internal resource

Axiom: $\forall a,\ (\exists\ r),\ requires\ (a, r) \wedge internal\_res(r) \equiv internal\_act(a)$ (FOL 62)

5.3 Resource and Activity Probing

Intuitions and concepts that lay the framework for a Theory of Resource Cost Units are presented in this section.

According to the Webster Dictionary, probing means "to examine or investigate; to conduct an exploratory investigation, search". Through resource and activity probing, we wish to conduct an exploratory investigation of resources and activities of an enterprise so as to search for activities that contribute costs to the resource cost units of resources required by other activities.

5.3.1 Intuitions about Resource Probing

Assume a resource is available to some activity. What is it that makes that resource available to the activity? Intuitively, it must be some past performed activities that makes the resource available to the activity.

Hence, based on this intuition and the meaning of probing, it is proposed that a resource be probed so as to investigate and identify *past performed activities* that have contributed towards making a resource available to some activity.

In the core TOVE ontology of activity, state and resource, the status of a state, and activity, is dependent on the status of the resources that the activity uses or consumes; and all states are assigned a status with respect to a point in time. Specifically, the commit status of an enabling state of an activity implies that resources have been reserved or made available for consumption or usage for an activity. Hence, more precisely, our intuition to probe a resource involves investigating and identifying activities that bring about the commitment of the probed resource to an activity. To commit a resource to an activity, the resource must be produced and reserved, or simply made available. Therefore, it is the cost of activities that participate in the production of the resource that contribute to the commit resource cost unit or commit_res_cost_unit.

Each identified activity of a resource probe itself requires resources, and each required resource itself may in turn be probed. The outputs of an activity may be products or services which may serve as resources to some other activity. Hence, the intuition to probe a resource, helps us identify some activity requiring resources, which in turn are the outcomes of some other activities, which in turn require resources, which in turn are the outcomes of some other activities, and so on. Thus a chain reaction of activities and resources is set in motion through the probe of a resource.

Further, an activity may require several different resources. However, from a cost perspective, some of these different required resources may have a negligible cost impact (say 10 percent or less) on the cost of the activity. The negligible cost impact level of 10 percent or less cited here may vary from company to company depending upon the circumstances. For example, for a company submitting tenders in a highly cost sensitive and competitive market may set a negligible cost impact level of say 3 percent or less in view of submitting bids in those market circumstances.

Hence intuitively, our notion of a *significant resource* is one that has a high imputed cost to attain and has an appreciable cost impact on the cost of an activity that requires the resource. Vice versa, an activity that requires at least one significant resource may be considered a significant activity which contributes a significant cost (say 10 percent or more, or some other level based on circumstances) towards the cost object for which the activity is performed. Therefore, for expediency and practical purposes of enterprise activity-based cost modelling, the probing of resources is limited to significant resources so as to seek out activities that contribute towards the cost of the probed resource to some overall level of reasonable accuracy and consistency that is meaningful to the decision maker. Reasonable accuracy and consistency may be best explained with Peter Turney's answer to the question, 'Is it better to be reasonably right, or precisely wrong?'. His answer is:-

> "Conventional cost systems often report the cost of products to fractions of a penny. For example, the cost of a product may be reported as $5.258637. Carrying product costing to such precision is a tribute to the power of computers and the accountant's traditional desire for exactness.
>
> It's a brave manager who challenges the accuracy of such a precise number. *But keep in mind that precision doesn't necessarily mean accuracy.* Computers always compute with great precision. But if you put in inaccurate numbers or use the wrong computational methodology, all you get is precision without accuracy.
>
> So how much should you trust the $5.258637 that your conventional cost system gives you? Too often the first digit is wrong. Worse yet, the decimal point is often in the wrong place too." [Turney 92]

5.3.2 Making a Case for Significant Activities in a Resource Probe

The domain of interest for this research is ABC/ABM. The probe of a resource uncovers several other activities and resources which in turn may be probed. To put resource probing in a practical perspective, how much data, detail and accuracy are required when resources are probed? The answer may be best expressed as Ruhl [97] stated:-

> "...... common misconceptions about ABC/ABM, include the following:
>
> - ABC requires massive amounts of data and detail;
> - ABC data must be extremely accurate; ......

> With regard to the accuracy and amount of data required, the needs of decision makers must be taken into account. Many decisions do not require extremely accurate (or large amounts of) data."

Further, the reasons to limit resource probing to significant activities fall within CAM-I's Cost Management System (CMS) Conceptual Design framework. These are [Berliner & Brimson 88]:

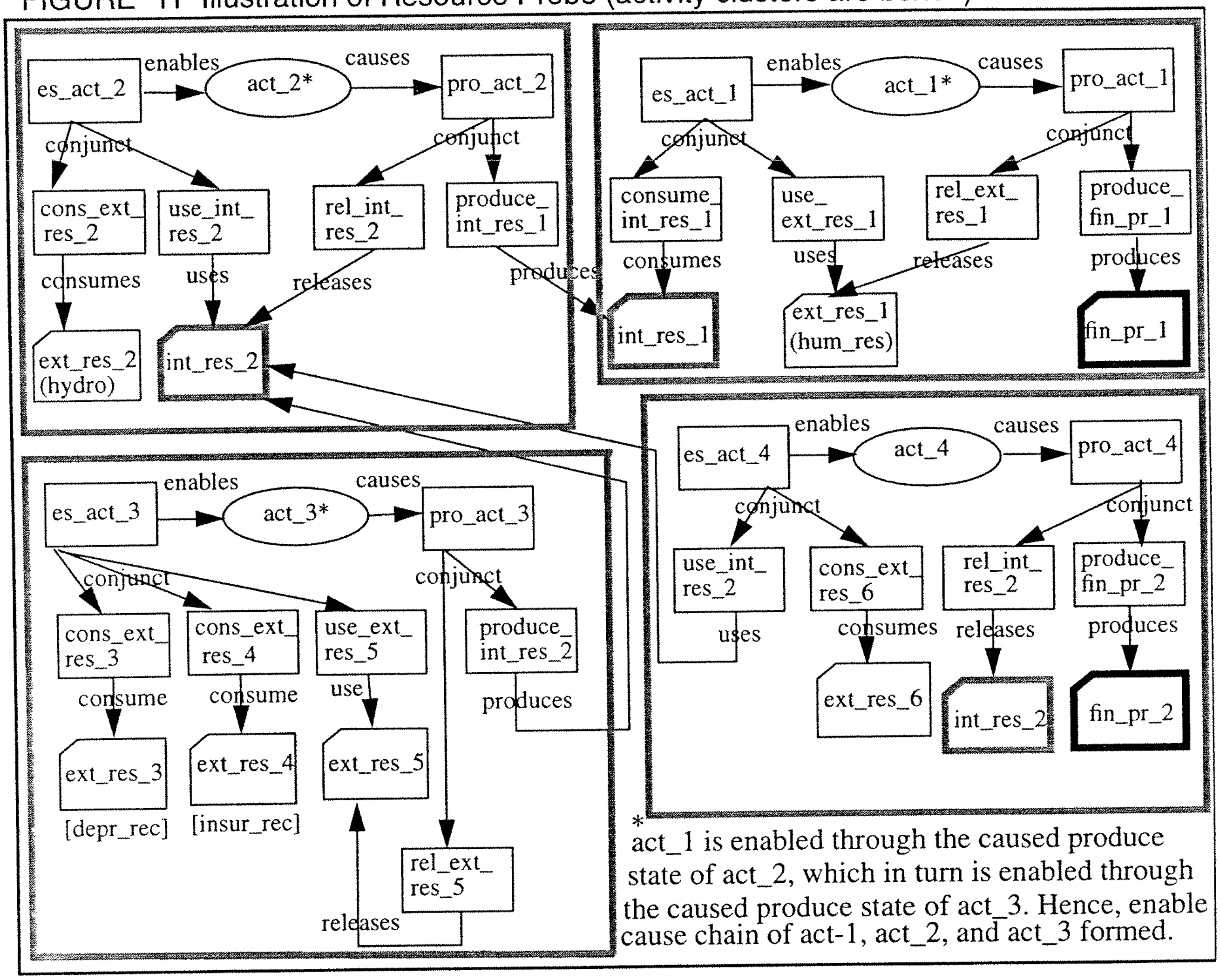
- Each company must define significant activities based on its business objectives and operating environment; CMS proposes to capture and track cost and performance data for only those few significant activities that constitute the bulk of the total work within the enterprise.
- Cost accounting can be viewed as identifying the cost of performing significant activities of the business. These activities are a common denominator amongst the three areas of cost accounting, performance measurement, and investment management. Activity accounting provides a logical framework for integrating these three critical areas of an enterprise.
- There are hierarchies of activities that cause other activities to occur and provide a basis for understanding the cost of performance drivers - a driver being an activity or condition that has a direct influence on the operational performance and cost structure, or both, of other activities.

Hence, for purposes of practicality and conformance to CAM-I's CMS framework, this research focuses upon *significant activities* and *significant resources* in building an Enterprise Model infrastructure for strategic cost management using TOVE's activity-cluster representation, where "an Enterprise Model is a computational representation of the structure, processes, information, resources, goals and constraints of a business, government activity, or other organizational system. It can be both definitional and descriptive - spanning what should be and what is. The role of an enterprise model is to achieve model-driven enterprise design, analysis and operation." [Gruninger & Fox 95]

5.3.3 Illustrating Resource Probe with the TOVE Representation

The nucleus of the TOVE representation is the activity cluster, that is, an activity along with its enabling and caused states is called an activity cluster. The state tree linked by an enables relation to an activity specifies what has to be true in order for the activity to be performed. The state tree linked to an activity by a causes relation defines what is true of the world once the activity has been completed. Resources required by an activity are linked to the enabling states of the activity through the consume or use relations, whereas resources (output) resulting from the completion of the activity are linked to the caused states of the activity through the produce or release relations. The outputs of an activity may be products or services which serve as resources to some other activity. Based on these fundamentals, Figure 11 depicts the chain reaction of activities and resources, or specifically activity clusters (shown boxed in Figure 11) in the TOVE representation that are set in motion through a resource probe.

FIGURE 11 Illustration of Resource Probe (activity clusters are boxed)



As illustrated in Figure 11, activity, act_1, consumes a resource, int_res_1. *The resource, int_res_1, is **probed** to seek the activities that make possible int_res_1 to be produced for act_1.* Obviously, int_res_1 must be brought to existence. As indicated, the activity, act_2, produces int_res_1. More precisely, the enabling state of act_1 is linked to the caused state of act_2, thereby forming an *enable_cause link* between act_1 and act_2. However, the activity, act_2, requires the resource, int_res_2. The resource, int_res_2, in turn is produced by the activity, act_3. We now have an *enable_cause link* between act_2 and act_3. Thus far, the two links form the *enable_cause chain* to consist of three activities sequenced as (act_3, act_2, act_1). In order to produce the resource, int_res_2, the activity, act_3, requires resources, ext_res_3 and ext_res_4 and ext_res_5.

For this illustration, assume that ext_res_3, ext_res_4 and ext_res_5 are resources that are supplied from sources external to the company modelled.

From the perspective of the TOVE representation, the probe of a resource involves identifying activities that bring about the commitment of the probed resource to an activity. To commit a resource to an activity, the resource must be made available or, more specifically in the language of the TOVE representation, the probed resource must be linked to the caused states of some other activity through the relation *produce.* Therefore, the cost of activities, be they "planning/service type activities" (e.g., engineering_design, production_scheduling, maintenance, client_request_processing, etc.) or "production/ manufacturing type activities" (e.g., product_fabrication, product_assembly, packaging_product, etc.) are part of the commit resource cost unit. The lucid activity cluster representation in TOVE clearly shows that *resource probing involves identifying some activity whose caused state produces the resource.*

In TOVE, the release of a resource by the caused state of an activity is represented through the release relation and represents that usage of the resource by the activity once completed is released for subsequent re-use. By definition in TOVE, the usage state of a resource implies that the resource can be re-used after its completion. The cost of producing some re-usable resource must be written off over a period of time or, equivalently, over the frequency of resource re-use by various activities for which the resource is anticipated or forecasted to be used. For example, a drug formula produced or developed by a pharmaceutical company may involve years of research and millions of dollars to be spent putting the cost of the formula to say $20 million. The formula is a re-usable resource to various production activities that produce the drug for the consumer market. In costing out the production activities that use the drug formula as a resource, in stands to reason that the cost of $20 million for the drug formula should be distributed over a forecasted or actual frequency of activity occurrences that re-use the formula, or for practical purposes, some estimated period of time over which the production activities are forecasted to or actually re-use the drug formula.

In the illustration of Fig. 11, act_2 and act_4 re-use the resource int_res_2. Assume that the cost of producing int_res_2 by the caused state of act_3 is distributed over some fixed time period which equivalently pro-rates an hour of usage of int_res_2 as being $100 per hour. If then an instance of act_2 requires int_res_2 for 3 hours whereas an instance of act_4 requires int_res_2 for 6 hours, then the int_res_2 resource cost contribution to activity costs of act_2 and act_4 would be $300 and $600 respectively. This example of Fig. 11 also shows that the probed resource int_res_2 links act_2 and act_4 through the *releases* and *uses* relationships; but the costs of act_2 and act_4 are independent of each other.

The illustration in Figure 11 clearly indicates that the probe of a resource leads to several layers of activities and resources. Therefore, in our quest for the commit resource cost unit, it is our intent to rationalize this deepening of the probe upto some *terminal frontier*, that is, some layer of activity clusters beyond which no more activity clusters are revealed for the enterprise through a resource probe. An activity in this terminal frontier will be referred to as a *frontier_activity*. Each resource required by a frontier_activity will be referred to as an *external resource*, that is, an extermal_resource is not produced by the caused state of some other activity, or simply put, an external_resource cannot be probed any further, and must be acquired directly from an outside or external source to the enterprise at some known specific *acquisition_unit_ price.*

Therefore, in the development of the Theory of Resource Cost Units, *this research assumes that an external resource need not be probed any further since it is a purchased resource and its paid acquisition cost, referred to as its acquisition_unit_price, is known.*

On the other hand, a resource that can be probed or equivalently, a resource that is produced by the caused state of some other activity is an *internal_ resource*. Then it stands to reason that *a frontier_ activity only requires external resources.* On the other hand, an *internal_activity requires at least one internal_resource.*

The *goal* of resource probing is to arrive at the terminal frontier, in a consistent manner based upon some principles, axioms and theories so that a "complete enterprise activity based model", referred to as Enveloped Activity Based Enterprise Model or EABEM by acronym, is attainable. The resource probing concept facilitates the trace of a resource, or stated differently, resource probing facilitates the traceability of activities to a specific resource.

In a sense, the layers of activity clusters (i.e., activities and resources) uncovered through the resource probing *envelope* or surround the probed resource. Hence, the concept of *commit envelope of a resource* is introduced in this research to represent the layers of activity clusters that make possible a resource being committed to an activity. From an activity representation standpoint, the commit envelope is considered an aggregate activity composed of or having for its sub-activities the several activities uncovered through the probe of the resource. Therefore, from a cost perspective, *the aggregate cost of the sub-activities that compose the commit_envelope of a resource contribute to the commit resource cost unit or the commit_res_cost_unit of the probed resource.*

5.4 Resource Probing and Other Envelopes for a Resource

The mutually exclusive and totally exhaustive status values for the state of a resource are commit, enabled, disenabled and reenabled. Our intuitions and discussions thus far have put forth the concept of resource probing and the commit_envelope of the resource as a means towards the commit_res_cost_unit of the resource associated with the commit status of the resource state.

We now explore the concept of resource probing and other possible "envelopes" of a resource towards each of the resource cost units. Other possible envelopes are explored by delving into the temporal behaviour of costs along the continuous time line of an activity instance in Figure 12. We do so by delving into the resource and activity costs in the course of an activity instance in order to gain further insight into the composition of resource cost units, the state status dependent resource cost contributions to the activity instance, and the status dependent activity costs that make up the cost of the activity instance.

5.5 Delving into Cost Behaviour during Instance of Activity

Activities exist forever, but primarily in a dormant or completed state. However, there elapses a finite period of time between the time-point of activity executing and the time-point of completion of each activity instance. In TOVE, we have recognized that the status of an activity instance changes with the passing of time due to the status value of the used or consumed terminal states associated with the resource required by the activity instance. The activity status may be dormant, executing, suspended, reExecuting or completed corresponding to the status value of the state for the required resource being committed, enabled, disenabled or re-enabled respectively, and overall completion of the activity,

5.5.1 Comments re: Resource Cost Units of a resource.

1. ***Committed resource cost unit:*** A resource that is committed to an activity may be viewed as "inventory committed to the activity". Money invested or sunk into the committed resource is not available for use in other areas of the enterprise, and, in fact, may have to be borrowed money. From a costing standpoint, the cost of borrowing the money or the cost of "foregone investment opportunity" from using this capital in other areas of the enterprise must be charged as the cost of capital (usually expressed as some percentage factor) against the activity to which the resource is committed [Plossl & Wight 68].

2. ***Enabled resource cost unit*:** A resource is usually committed to an activity for future use. We consider an instance of an activity as being instantiated at the time point when a resource is committed to the activity. The committed resource enables the activity to execute. At the time point when the activity begins execution, the enabled resource is used or consumed by the enabling states of the executing activity. From a cost perspective, the enabled resource cost unit metric is taken to be equivalent to the committed resource cost unit metric as each unit of resource required by the executing activity costs an amount equal to its commit resource cost unit.

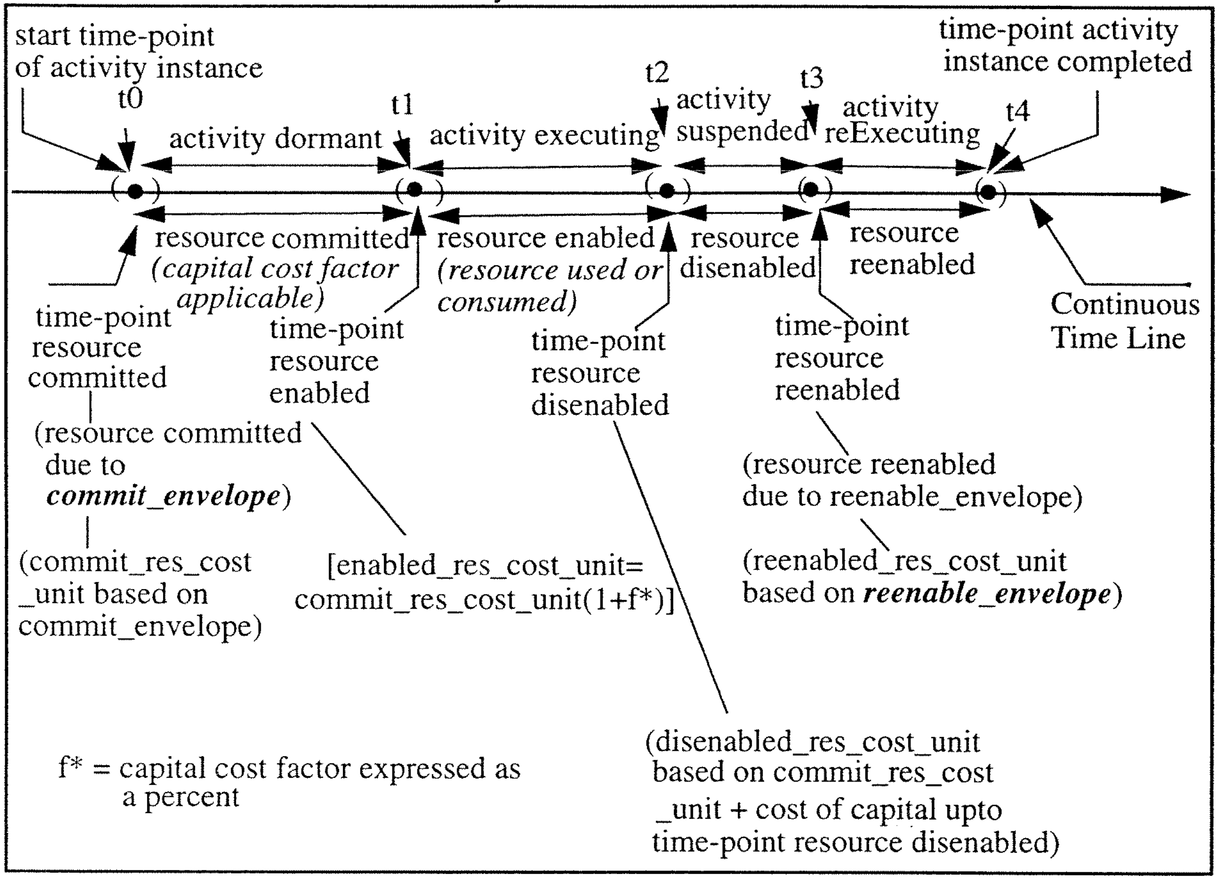
3. ***Disenabled resource cost unit*:** A resource that becomes disenabled brings about the suspension of an executing activity that requires it. Though the cost value of the resource has not changed in this state, the enterprise loses opportunities during the time period the activity is suspended due to the disenabled resource. While the resource is disenabled, we must consider the notion of lost opportunity cost, viz., the return that could have been realized if the resource state was enabled and the activity executing. Hence, from a costing standpoint, a lost opportunity cost factor (usually expressed as some percentage factor) must be taken into consideration when computing the disenabled resource cost for an activity that has been suspended due to the disenabled resource.

4. ***Reenabled resource cost unit:*** Typically, a disenabled resource is reenabled by the execution of activities that bring about the "repair" of the disenabled resource so that the suspended activity due to the disenabled resource may resume execution. Therefore, we consider the cost value of a reenabled resource as being greater than that of the initial enabled resource simply because the cost of "repair" activities must be sunk into the disenabled resource before it is reenabled. An enterprise may consider cumulatively incrementing the value of the reenabled resource cost unit with each iteration that a resource is disenabled and then reenabled.

5.5.2 Developing Axioms for Resource Cost Units of a Resource

The comments regarding the Resource Cost Units of a resource give direction to the development of axioms that help us deduce resource cost units

FIGURE 12 Illustration: Activity Instance on Continuous Time Line



In Figure 12, assume that a resource is required by an instance of an activity, and the status of the resource state and activity are as follows along the time line intervals:-

- resource committed, activity instantiated at t0;
- resource committed, activity dormant for [t0, t1);
- resource enabled, activity executing for [t1, t2), resource used or consumed by activity;
- resource disenabled, activity suspended for [t2, t3);
- resource reenabled, activity reExecuting for [t3, t4];
- activity terminated at t4.

Between t0 and t1 no action occurs, only time elapses. During the committed period, capital invested in the committed resource is "sitting idle", i.e., it is not generating any type of interest earnings or returns that could have been earned from other opportunities. In general, returns on capital invested will vary based on several factors such as principal amount invested, time duration money remains invested, and the risks associated with the investment vehicle.

During the dormant activity interval, the enterprise must bear the cost of capital tied or sunk into the committed resource. For example, longer the dormant activity interval, the higher the interest cost will be on borrowed funds expended on the committed resource. *Once a resource is committed to an activity, the costs of the resource are attributed to the activity.* The cost of the dormant activity is intuitively related to the cost of capital available to the enterprise. The cost of funds to the enterprise is referred to as the enterprise's cost of capital, a cost treated as a given for purposes of this research. This research is NOT about optimizing returns on capital or the alternative investments for capital or the formalization of the cost of capital. *We treat the cost of capital as an exogenously determined variable and define cost of capital simply as a given percentage per annum (cc_rate) for cost of funds available to the enterprise. We term the pro-rated percentage applicable to the resource committed period as the capital_cost_factor*.*

[* Note: The capital_cost_factor term and the notation ccf used in this research should *not* be mistaken for the "capital cost factor or ccf" concept of Engineering Economics used for calculating depreciated value of acquired capital goods based upon "class of goods" as defined for tax purposes.]

For example, a company X will have a capital_cost_factor lower than that of company Y if X's credit rating with a bank is better than that of Y. In this instance, for example, in interval [t0, t1), X may have a capital_cost_factor based upon 6 percent per annum, whereas Y may have a capital_cost_factor based upon 9 percent per annum. In this case, assuming that [t0, t1) represents a time interval of 4 months, then the capital_cost_factor for X would be 2 percent, and that for Y would be 3 percent, each being the pro-rated percentage applicable to the resource committed period for 4 months.

Let us now use the capital_cost_factor of 2 percent for company X to compute the committed resource cost and dormant activity cost for the committed resource interval or dormant activity interval of 4 months. Assume 1 unit of resource R1 is the only resource required and consumed by activity A1. Assume R1 is an external_resource acquired at a cost of $10,000 per unit with borrowed funds at the interest rate of 6 percent per annum. The committed_res_cost_unit would be $10,000 as it is an external_resource. The committed resource cost or committed_res_cost for R1, or in this example, the equivalent dormant activity cost for A1 would be 2 percent of $10,000 or $200. In this case, the dormant activity cost (dormant_act_cost) or the committed resource cost (committed_res_cost) of $200 may be looked upon as real costs sustained by company X as part of the "cost of borrowing" total in the general ledger for X. Note that, if the resource committed period or the activity dormant period of [t0, t1) is reduced to a null period, the dormant_act_cost or the committed_res_cost would be zero. In essence, just-in-time (JIT) planning strives to reduce or minimize the resource committed period. This example illustrates that activity based costs are real and help reveal activities and their corresponding costs against items of the general ledger for a company. These revelations of activities and their costs provide more pertinent information to decision makers involved in re-engineering processes.

Therefore, based upon the foregoing discussion, this research formalizes the precondition for the cost of capital rate, denoted as cc_rate, to exist in a situation σ.

This precondition is that a resource r must be committed to an activity in that situation σ. We formalize this precondition for a cost of capital rate in Axiom (FOL 63) below. Further, we formalize Axiom (FOL 64) in FOL to deduce the capital cost factor (ccf by notation) expressed as a pro-rated percentage based upon the duration of the committed interval of a resource to the activity and the applicable cc_rate. [Note that Axiom (FOL 6) of Ch. 4 repeated below defines the committed interval of a resource to an activity].

Ch.4: Axiom (FOL 6): $\forall a, s, r, t, t'$, ***committed_interval*** $(a, r, t, t') \equiv \forall t'', t \leq t'' < t' \supset$ ***holdsT*** $(committed(s, a), t'') \wedge \neg$ ***holdsT*** $(committed(s, a), t') \wedge$ ***holdsT*** $(committed(s,a), t)$

Axiom: $\forall \sigma$, ***r, cc_rate, poss (cost_of_capital (cc_rate, r),*** σ) $\supset \exists$ ***a,***
holds (committed (r, a), σ) **(FOL 63)**

Axiom: $\forall$***a, r, cc_rate, t0, t1, t0 < t1, committed_interval (a, r, t0, t1)*** $\wedge$ ***holdsT (cost_of_capital (cc_rate, r), t0)*** $\wedge$ ***holdsT (enabled (s, a), t1)*** $\supset \exists$ ***ccf, capital_cost_factor (ccf, r, t0, t1)*** $\wedge$ ***ccf = [(cc_rate) . (t1 - t0)]*** **(FOL 64)**

The defined capital_cost_factor ccf is then used to deduce the committed_res_cost for the resource. The capital cost factor modifies Axiom (FOL 10) and Axiom (FOL 11) of Ch. 4 to Axiom (FOL 65) and Axiom (FOL 66) respectively. Axioms (FOL 10) and (FOL 11) of Ch. 4 are repeated below for convenience. In Ch. 4, capital cost factor was not considered.

For a committed state resource required by an activity in *continuous mode* over a time interval [t, t']:-

Ch. 4: Axiom (FOL 10): $\forall$*a, r, c, v, ccf, t, t', t", t* $\leq$ *t"* $\leq$ *t', committed_interval (a, r, t, t')* $\wedge$ *continuous_mode (r, a)* $\wedge$ *[use_spec (r, a, t, t', q, rate, unit)* $\vee$ *consume_spec (r, a, t, t', q, rate, unit)]* $\supset$ *[committed_res_cost (a, r, c, t")* $\equiv$ *committed_res_cost_unit (a, r, q, v)* $\wedge$ *c = v . (t" - t) . rate]*

Axiom: $\forall$***a, r, c, v, ccf, t, t', t", t*** $\leq$ ***t"*** $\leq$ ***t', committed_interval (a, r, t, t')*** $\wedge$ ***continuous_mode (r, a)*** $\wedge$ ***[use_spec (r, a, t, t', q, rate, unit)*** $\vee$ ***consume_spec (r, a, t, t', q, rate, unit)]*** $\wedge$ ***capital_cost_factor (ccf, r, t, t')*** $\supset$ ***[committed_res_cost (a, r, c, t")***
$\equiv$ ***committed_res_cost_unit (a, r, q, v)*** $\wedge$ ***c = v . ccf. (t" - t) . rate]*** **(FOL 65)**

For a committed state resource required by an activity in *discrete mode* over a time interval[t, t']:-

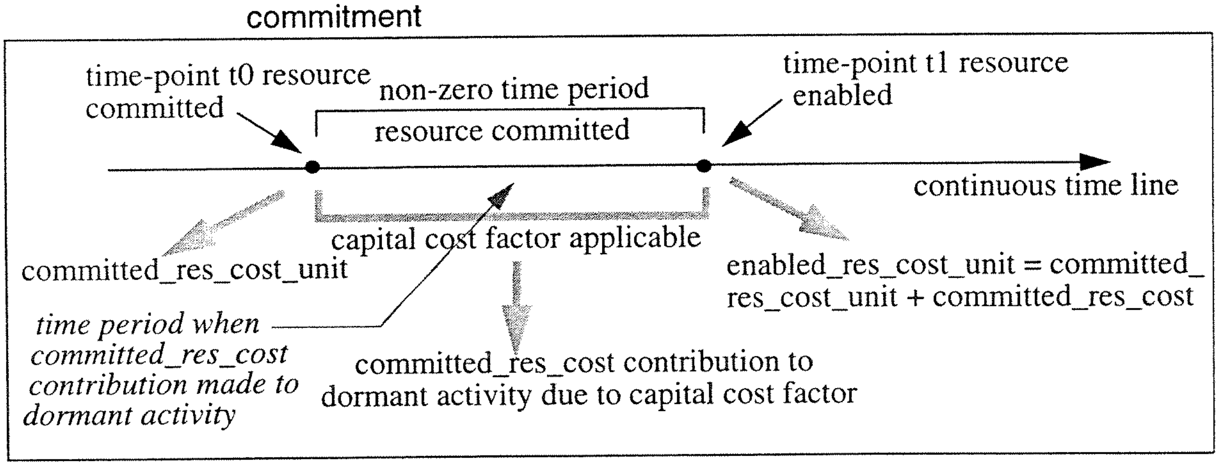
Ch. 4: Axiom (FOL 11): $\forall$*a, r, c, v, ccf, t, t', t* $\leq$ *t"* $\leq$ *t', committed_interval (a, r, t, t')* $\wedge$ *discrete_mode (r, a)* $\wedge$ *[use_spec (r, a, t, t', q, rate, unit)* $\vee$ *consume_spec (r, a, t, t', q, rate, unit)]* $\supset$ *[committed_res_cost (a, r, c, t")* $\equiv$ *committed_res_cost_unit (a, r, q, v)* $\wedge$ *c = v . q]*

Axiom: $\forall a, r, c, v, ccf, t, t', t \leq t'' \leq t'$, *committed_interval* $(a, r, t, t') \wedge$ *discrete_mode* $(r, a) \wedge$ [*use_spec* $(r, a, t, t', q, rate, unit) \vee$ *consume_spec* $(r, a, t, t', q, rate, unit)$] $\wedge$ *capital_cost_factor* $(ccf, r, t, t') \supset$ [*committed_res_cost* $(a, r, c, t'') \equiv$ *committed_res_cost_unit* $(a, r, q, v) \wedge c = v \,.\, ccf.\, q$] (FOL 66)

At time-point t1 the activity begins executing and the resource is enabled. The resource is consumed or used while the status of the activity is executing. The $ value of 1 unit of resource at the time-point the resource is enabled is termed enabled_res_cost_unit. With the passing of time from t0 to t1 and, as is the case of a principal amount earning interest with time in a bank account, the committed resource cost at t0 increases with the passing of time from t0 to t1 based upon the capital_cost_factor ccf applicable to the resource committed period. The enabled_res_cost_unit at t1 is the value of the commit_res_cost_unit plus the increase to the committed resource cost between t0 and t1 due to the capital_cost_factor (see Figure 13). The computation for the enabled_res_cost_unit of a resource may be axiomatized as follows:-

Axiom: $\forall a, s, r, q, v_1, ccf, t0, t1, t0 < t1$, *committed_interval* $(a, r, t0, t1) \wedge$ *holdsT* (*enabled* $(s, a), t1) \wedge$ *capital_cost_factor* $(ccf, r, t0, t1) \wedge$ *committed_res_cost_unit* $(a, r, q, v_1) \supset \exists\, v_2$, *enabled_res_cost_unit* $(a, r, q, v_2) \wedge [\, v_2 = v_1 \,.\, (1 + ccf)\,]$ (FOL 67)

FIGURE 13 enabled_res_cost_unit: non-zero time period of resource commitment

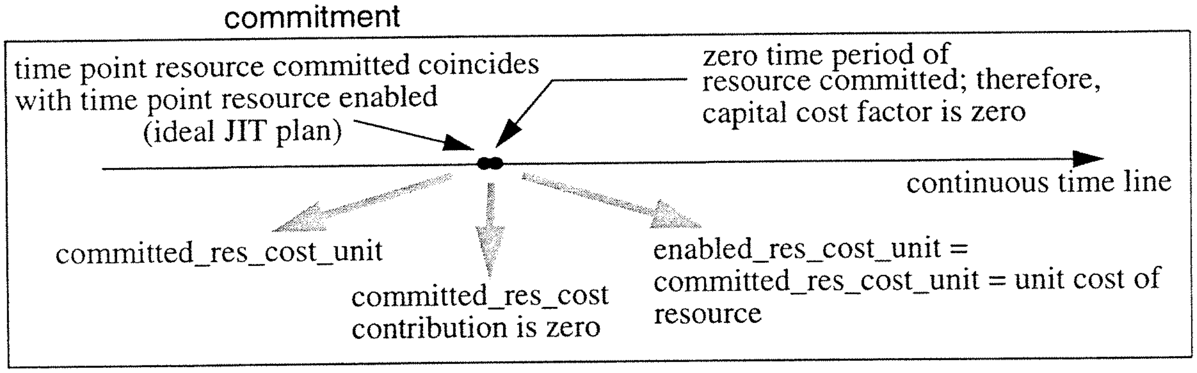


On the other hand, if the period of resource commitment is "short lived" or theoretically, a null period (see Figure 14), there is no enhancement potential of the commit_res_cost_unit due to no capital cost factor being applicable to the commit period. Hence, we state the following axiom:-

Given that there is no time period for which a resource has been committed to an activity, then the enabled resource cost unit of the resource is equivalent to its committed resource cost unit.

$$\textbf{\textit{Axiom:}}\ \forall\ (a, s, r, q, v1), \neg\exists\ (t0, t1),\ committed\_interval(a, r, t0, t1) \wedge committed\_res\_cost\_unit\ (a, r, q, v1) \supset \exists\ v2,\ enabled\_res\_cost\_unit\ (a, r, q, v2) \wedge [v2 = v1] \quad \text{(FOL 68)}$$

FIGURE 14 enabled_res_cost_unit: zero time period of resource commitment



As shown in Figure 12, the activity instance is executing during the time interval [t1, t2), and the resource that was earlier committed to the activity due to its requirements is used or consumed. The cost of the used or consumed resource may be deduced through Axioms (FOL 12) and (FOL 13) of Ch. 4. These axioms are repeated below for the reader's convenience:-

For an enabled state resource required by an activity on a *continuous basis* over a time interval [t, t']:-

Ch. 4 Axiom (FOL 12): $\forall a, r, c, v, t, t', t'', t \le t'' \le t'$, *enabled_interval* $(a, r, t, t') \wedge$ *continuous_mode* $(r, a) \wedge [$*use_spec* $(r, a, t, t', q, rate, unit) \vee$ *consume_spec* $(r, a, t, t', q,$

rate, unit)] $\supset$ *[enabled_res_cost (a, r, c, t")* $\equiv$ *enabled_res_cost_unit (a, r, q, v)* $\wedge$ *c = v.(t" - t).rate]*

For an enabled state resource required by an activity on a *discrete basis* over a time interval [t, t']:-

Ch. 4: Axiom (FOL 13): $\forall$*a, r, c, v, t, t', t* $\leq$ *t"* $\leq$ *t', enabled_interval (a, r, t, t')* $\wedge$ *discrete_mode (r, a)* $\wedge$ *[use_spec (r, a, t, t', q, rate, unit)* $\vee$ *consume_spec (r, a, t, t', q, rate, unit)]* $\supset$ *[enabled_res_cost (a, r, c, t")* $\equiv$ *enabled_res_cost_unit (a, r, q, v)* $\wedge$ *c = v.q]*

As stated in Axiom (FOL 31) of Ch. 4, the cost of the executing activity is based upon the sum of costs of enabled resources that are used or consumed by the executing activity.

Continuing along the time line of Figure 12, we assume the resource becomes disenabled and the activity is suspended at t2. Given that the resource is required by the activity for all situations s1, intuitively it follows that the disenabled resource must be "repaired" so that the resource is renabled and the activity may re-execute. In a sense, the resource is now a repairable resource of the activity. Once the resource is repaired, it is possible to execute the activity. The relationship between the resource r and the activity a may be considered as being repairable, i.e., repairable (r, a).

We define the relationship **repairable (r, a)** between resource r and activity a as implying that a requires r, and for all situations s1, there exists some other situation s2 that follows s1 so that a is possible in situation s2. We formalize this relationship in FOL as follows:-

Axiom: $\forall$***a, r, s1, repairable(r, a)*** $\supset \exists$ ***s2, requires (a, r)*** $\wedge \neg$ ***do (a, s1)*** $\leq$ ***s2*** $\wedge$ ***poss(a,s2)*** **(FOL 69)**

Intuitively, the situation referred to as s2 is brought about by "repairing" or reenabling the disenabled resource r which is required by activity a; and therefore, it is possible for activity a to resume execution or to re-execute with the reenabled resource. From the perspective of situation calculus [McCarthy & Hayes 69], all situational changes are the result of some identified actions. It is these actions that bring about situation s2 whereby the disenabled resource of situation s1 is reenabled so that it is possible to execute the activity a in situation s2. We consider these actions that bring about the reenablement of the resource as being sub-actions of an aggregate action that we term the "reenable envelope". Similar to the previous discussions on the commit envelope, the **reenable envelope** is considered an aggregate activity that consists of identified or traced activities or sub-actions that are completed so that the disenabled resource is reenabled. For purposes of costing and similar to considerations for the resources required by the sub-actions of the commit envelope, each of the resources required by the sub-actions of the reenable envelope are probed until frontier activity clusters, viz., frontier activities together with their external resources, are identified. Therefore, from a cost perspective, *the aggregate cost of the completed activities that compose the reenable_envelope of a resource contribute to the reenable resource cost unit or the reenable_res_cost_unit of the disenabled resource.*

An attempt is now made to answer the following: **What is the cost of a Suspended Activity?**

Assuming that the sub-actions of the reenable_envelope are executed during the time period [t2, t3), the disenabled resource is reenabled at t3 (Figure 12). At t3, the activity begins to reexecute. During the period [t2, t3), the activity remains suspended.

From an activity cost standpoint, what is the cost of the suspended activity? Assumptions are made in order to answer this question. These assumptions are:-

1. An activity is suspended when at least one of its required resources becomes disenabled.
2. The change of resource state status from enabled to disenabled is instantaneous. Owing to this *instantaneous* change, there is no change due to cost of money in the dollar value of the resource itself when the status of the activity instance changes from executing to suspended. In other words, the disenabled resource cost unit at t2 is considered equivalent to the enabled resource cost unit while the activity was executing just prior to the activity being suspended. We state this assumption as an axiom as follows:-

Axiom: $\forall$ *(a, r, q, s, v2, v3, t1, t2), t1 < t2, execute_act_interval (a, t1, t2)* $\wedge$ *requires(a, r)* $\wedge$ *holdsT(enabled (s, a), t1)* $\wedge$ $\neg$ *holdsT(enabled (s, a), t2)* $\wedge$ $\neg$ *holdsT (completed (a), t2)* $\wedge$ *enabled_res_cost_unit (a, r, q, v2)* $\supset$ *holdsT(disenabled(s, a), t2)* $\wedge$ *disenabled_res_cost_unit (a, r, q, v3)* $\wedge$ *(v3 = v2)* (FOL 70)

3. During the period [t2, t3) when the activity instance is suspended, one has the intuition that the enterprise is "losing opportunities", that is, the suspended activity due to the disenabled resource is costing the company money based upon some "opportunity cost", viz., some rate of return on some known or anticipated payment that is foregone by the enterprise on account of the activity being suspended due to the disenabled resource. Therefore, in essence, the suspended activity conjures some rate of return the enterprise demands for accepting a payment delayed by the activity suspension time period between t2 and t3.

4. In dealing with opportunity cost associated with the suspended activity, this research limits its scope to a discount factor associated with the present value of any delayed payoff [Brealey et al. 97] based upon the well accepted cost accounting principle:

   "a dollar today is worth more than a dollar tomorrow, because the dollar today can be invested to start earning interest immediately."

   According to Brealey et al. [97], the present value of a delayed payoff may be found by multiplying the payoff by a discount rate that is less than one. If the discount rate were more than one, a dollar today would be worth less than a dollar tomorrow. If C denotes the expected payoff at time period 1 year hence, then

   Present value (PV) = (discount rate) x (C) ....equation (i)

   Given that the rate of return rr, expressed as a percentage per year, is the reward the enterprise demands for accepting delayed payment, then the discount factor rate may be expressed as the reciprocal of one plus the rate of return:

   Discount rate = 1/ (1 + rr) ....equation (ii)

An activity loses "opportunities" when suspended due to a required resource being disenabled. Intuitively, an "opportunity cost" is associated with the activity being suspended. It is customary to account for the "opportunity cost" through an opportunity cost factor, say p, where p is the percentage of the discount rate pro-rated over the time period the activity is losing "opportunities", that is, being suspended. Hence, from equations (i) and (ii), given that the suspended activity interval is [t2, t3), then:

Opportunity cost factor = (Discount rate) x (t3 - t2) .... (iii)

Therefore, based upon the foregoing assumptions and discussions, this research formalizes the precondition for the possibility of a rate of return, r_rate, and the possibility of discount rate, d_rate, to exist in a situation σ. This precondition is that an activity be suspended in that situation σ due to at least one of its required resources being disenabled. The time interval (t, t') for which the resource is disenabled has been defined as the disenabled interval in Ch. 4 with Axiom (FOL 8) restated below:-

Ch. 4: Axiom (FOL 8): $\forall a, s, r, t, t'$, ***disenabled_interval*** $(a, r, t, t') \equiv \forall t'', t \leq t'' < t' \supset$ ***holdsT*** *(disenabled (s, a), t'')* $\wedge \neg$ ***holdsT****(disenabled (s,a), t')* $\wedge$ ***holdsT*** *(disenabled (s, a), t)*

Given that an activity is suspended due to its required resource being disenabled, we assume that there exists an annualized rate of return, denoted r_rate expressed as a percentage, foregone by the enterprise in that situation. This is stated in FOL as follows:-

Axiom: $\forall a, \exists (r, \sigma)$, ***holds (requires (a, r), σ) $\wedge$ holds (disenabled (r, a), σ) $\supset$***
$\exists$ r_rate, holds(rate_of_return(r_rate, a), σ) (FOL 71)

Given that the annualized rate of return applicable in a situation is r_rate, there exists a discount rate, denoted d_rate, applicable in that situation that is equivalent to the reciprocal of one plus the rate of return. [Equation (ii)]. This is stated in FOL as follows:-

Axiom: ***$\forall$(a, r, σ, r_rate), holds(rate_of_return(r_rate, a), σ) $\supset \exists$ d_rate, holds(discount_rate(d_rate, a), σ) $\wedge$ [d_rate = 1/(1 + r_rate)]*** (FOL 72)

The opportunity cost factor p expressed in equation (iii) is then computed according to the following axiom:-

Given that a resource is disenabled for a time period [t2, t3), the applicable annualized discount rate is d_rate for that period, and the resource is reenabled at time t3; then the opportunity cost factor p applicable to the period is equivalent to the discount rate pro-rated over the time period (t2, t3). We state this axiom in FOL as follows:-

Axiom : $\forall$***a, r, d_rate, t2, t3, t2 < t3, disenabled_interval (a, r, t2, t3)*** $\wedge$ ***holdsT (discount_rate (d_rate, a), t2)*** $\wedge$ ***holdsT (reenabled (s, a), t3)*** $\supset \exists$ ***p, opportunity_cost_factor (p, a, t2, t3)*** $\wedge$ ***p = [(d_rate) . (t3 - t2)]*** **(FOL 73)**

The defined opportunity cost factor p is a percentage number (e.g., 25% = 0.25) and is then used to deduce the disenabled_res_cost for each resource required by an activity. In Ch. 4, opportunity cost factor was not considered in Axiom (FOL 14) and Axiom (FOL 15). However, if opportunity cost factor is considered, Axiom (FOL 14) and Axiom (FOL 15) would be modifed to Axiom (FOL 74) and Axiom (FOL 75) respectively. Axiom (FOL 14) and Axiom (FOL 15) of Ch.4 are repeated here for convenience:-

[Note: (i) The computation of p itself is time dependent; and it is this percentage that is multiplied by the "notional cost" of resource that would have been used if the activity were not suspended. In other words, the cost of loss opportunity due to a suspended activity is expressed as a percentage of the cost of resource that would have been used if the activity were not suspended. (ii) It is assumed that the disenabled resource is not perishable. If the disenabled period [t2, t3] is longer than the "spoil time" of the resource, the entire cost of the "spoilt resource" should be borne by the suspended activity. In other words, the opportunity cost factor should be at least 100% or 1.00].

For a resource required by an activity in *continuous mode* over a suspended activity time interval [t, t']:-

Ch. 4: Axiom (FOL 14): $\forall$ *(a, r, c, v, p, t, t', t"),* $t \leq t" \leq t'$*, suspended_act_interval (a, t, t')* $\wedge$ *continuous_mode (r, a)* $\wedge$ *[use_spec (r, a, t, t', q, rate, unit)* $\vee$ *consume_spec (r, a, t, t',* *q, rate, unit)]* $\supset$ *[disenabled_res_cost (a, r, c, t")* $\equiv$ *disenabled_res_cost_unit (a, r, q, v)* $\wedge$ *c = v .(t" - t).rate]*

Axiom: $\forall$ **(a, r, c, v, p, t, t', t"), t** $\le$ **t"** $\le$ **t', suspended_act_interval (a, t, t')** $\wedge$ **continuous_mode (r, a)** $\wedge$ **[use_spec (r, a, t, t', q, rate, unit)** $\vee$ **consume_spec (r, a, t, t', q, rate, unit)]** $\wedge$ **opportunity_cost_factor (p, a, t, t')** $\supset$ **[disenabled_res_cost (a, r, c, t")** $\equiv$ **disenabled_res_cost_unit (a, r, q, v)** $\wedge$ **c = p . {v .(t" - t).rate}]** **(FOL 74)**

[Note: in Axiom (FOL 14), if the resource was not disenabled, the cost of resource requirement in continuous mode would have been {v.(t"- t). rate}].

For a resource required by an activity in *discrete mode* over a suspended activity time interval [t, t']:-

Ch. 4: Axiom (FOL 15): $\forall$ *(a, r, c, p, t, t', t"), t* $\le$ *t"* $\le$ *t', suspended_act_interval (a, t, t')* $\wedge$ *discrete_mode (r, a)* $\wedge$ *[use_spec (r, a, t, t', q, rate, unit)* $\vee$ *consume_spec (r, a, t, t', q, rate, unit)]* $\supset$ *[disenabled_res_cost (a, r, c, t")* $\equiv$ *disenabled_res_cost_unit (a, r, q, v)* $\wedge$ *c = v.q]*

Axiom: $\forall$ **(a, r, c, p, t, t', t"), t** $\le$ **t"** $\le$ **t', suspended_act_interval (a, t, t')** $\wedge$ **discrete_mode (r, a)** $\wedge$ **[use_spec (r, a, t, t', q, rate, unit)** $\vee$ **consume_spec (r, a, t, t', q, rate, unit)]** $\wedge$ **opportunity_cost_factor (p, a, t, t')** $\supset$ **[disenabled_res_cost (a, r, c, t")** $\equiv$ **disenabled_res_cost_unit (a, r, q, v)** $\wedge$ **c = p . { v.q }]** **(FOL 75)**

[Note: in Axiom (FOL 14), if the resource was not disenabled, the cost of resource requirement in continuous mode would have been {v.q}]

Our discussion towards answering the question - what is the cost of a suspended activity? - has dealt with the computation of disenabled resource cost unit, and has also considered one's intuitions about "opportunity costs" when an activity is suspended. Hence, from an ABC standpoint, we are able to deduce the cost of a suspended activity due to at least one of its resources being disenabled by computing the cost contributions of each of its required resources through the disenabled_res_cost axioms [Axiom (FOL 14) and Axiom (FOL 15)]; and summing these cost contributions towards the suspended activity cost as was axiomitized in Axiom (schema) (FOL 32) of Ch. 4 (repeated below):-

Ch. 4: Axiom (schema) (FOL 32): For each activity, a, we have an axiom of the form: $\forall$ $c, c', r_1, r_2, ... r_n, t,$ *suspend_act_cost* $(a, c', t) \equiv$ [*total_disenabled_res_cost* $(a, r_1, c_1, t) \wedge$ *total_disenabled_res_cost* $(a, r_2, c_2, t) \wedge$ *total_disenabled_res_cost* $(a, r_3, c_3, t) \wedge .... \wedge$ *total_disenabled_res_cost* $(a, r_n, c_n, t)] \wedge c' = c_1 + c_2 + ... + c_n$

[Note: If the suspended activity cost incurred by the company is high due to frequent resource breakdowns or unduly long repair times for the disenabled resource, the management should consider major overhaul of the resource, or should consider acquiring a "new" resource. In other words, suspended activity costs and disenabled resource costs should be monitored for "capital goods acquisition".]

<u>Section Conclusion:</u> Having examined the cost behaviour for an activity instance, we conclude that the committed resource cost unit leads us to deducing the enabled resource cost unit and the disenabled resource cost unit of a resource. The committed resource cost unit should be computed based upon the concept of a commit_envelope of activities due to Resource Probing. In like manner, the reenable resource cost unit should be computed based upon the reenable_envelope of activities due to Resource Probing of a disenabled resource. Hence, in order to deduce resource cost units of a resource, it is sufficient that we establish the deduction of the committed resource cost unit of a resource.

5.5.3 Activity Probing and Cost of Specifying an Activity

Much of the literature on ABC and ABM does not address the issue about the cost of specifying an activity. By this we mean, what is the cost of making possible the occurrence or existence of an activity as we observe its execution. For example, suppose there exists a heart_examination activity in a hospital. We implicitly understand that for this activity to actually occur, there must have been some activity specification prepared for this activity to manifest itself in the real world. From the perspective of industrial engineering, a typical specification of an activity informs us of resources required, methodology of execution, and a time estimate to complete the activity. For example, it is not uncommon to have such activity specifications visibly displayed in hard copy form at manufacturing work-stations assembling computers; or at banks, where activity specifications may be in digital form to be accessed through a computer terminal.

But the preparation or production of this activity specification itself calls for various activities to be completed. In other words, an activity performed at the present time, must have had a set of activities completed in the past that specifies the activity. Similar to the concept of Resource Probing that traces a set of activities that contribute to a resource being there in the real world, the concept of Activity Probing traces a set of activities that specifies an activity manifesting itself in the real world. It logically follows that the cost of specifying an activity, say X, is the aggregate cost of a set of activities that *specifies* activity X. Hence the need for Activity Probing towards costing.

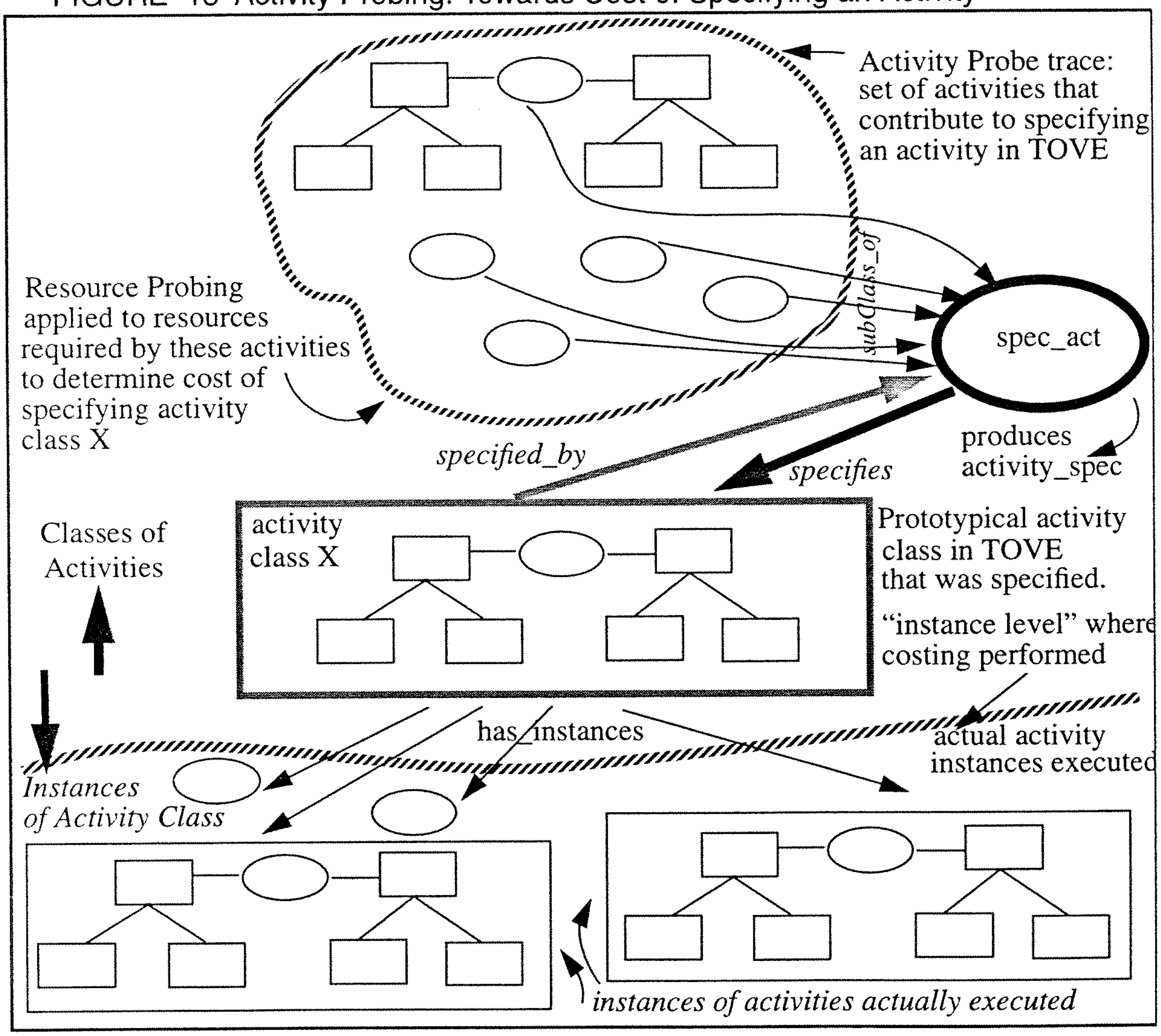
More importantly, the concept of Activity Probing points the way towards tracing classes of "overhead activities" (as is commonly referred to in companies) such as research & development, marketing, engineering, product designing, employee training and human resource planning, just to name a few of the "overhead activities" that contribute towards formulating a specification for an activity. Frequently, "overhead activities" are sub-activities of the "specification activity" that specifies an activity. Why is this issue important to us? The answer to the question may be best expressed as follows:-

- "The management of costs - and particularly overhead - has become one of the main business preoccupations. In this situation the starting point is to determine which costs are product-driven and which are really overhead. ...... The need is to understand what causes cost and to trace the relationships between costs, processes and products. One has to cost the activities." [Sheridan 89]

Therefore, the concept of Activity Probing as put forth in this research is premised upon the notion that it is a set of activities that contribute towards an activity being specified (Fig. 15). From an activity modelling standpoint, the set of activities responsible for specifying an activity are considered subClasses of a "specification activity", denoted as spec_act in Fig. 15. From the TOVE modelling perspective, the specification activity produces an "activity specification", (activity_spec) in Fig. 15. Activity Probing seeks to identify those activities that contribute towards the making of an activity specification.

Activities identified through Activity Probing are considered subClasses of the specification activity. These subClass activities require resources that are probed (Resource Probing) to determine the costs of the resources, so that each subClass activity is costed. The cost of the specification activity (spec_act) is the sum of the cost of its subClass activities. The cost of the activity specification (activity_spec) is the activity cost of spec_act.

FIGURE 15 Activity Probing: Towards Cost of Specifying an Activity



Definition: A specification activity, denoted spec_act, is a class of activity.

Axiom: $\forall$ ***(spec_act), activity (spec_act)*** (FOL 76)

In Fig. 15, it is clearly illustrated that the spec_act is related to an activity through the *specifies* relationship. We formally define the *specifies* relationship as follows:

The ground term *specifies* is a relation between two classes of activities. Costing of an activity is done at the instance level as indicated in Fig. 15. From the perspective of costing, specifies is a relationship between an activity and its specification activity. We state this as an axiom as follows:-

Definition: For each activity a, there is some activity, denoted spec_act, that ***specifies*** it.

Axiom: $\forall$ ***(a, s),*** $\exists$ ***spec_act, holds(specifies (spec_act, a), s)*** (FOL 77)

Given that an activity, spec_act, specifies another activity, implies that the former produces the activity specification (denoted activity_spec) of the latter. Stated as an axiom in FOL, we have:-

Axiom: $\forall$ ***(a, spec_act, s), holds(specifies(spec_act, a), s)*** $\equiv \exists$ ***activity_spec, produces(spec_act, activity_spec)*** $\wedge$ ***requires(a, activity_spec)*** (FOL 78)

The cost of specifying an activity is the cost of its specification activity. The cost of the specification activity is the aggregation of activity costs for each of its subClass activities (identified by Activity Probing). The cost of the class activity, spec_act, may be costed according to Axiom (FOL 40) of Ch. 4., and re-stated below:-

Ch. 4: Axiom (schema) (FOL 40): For each Class activity, a'_{ix}, the axiom is of the form:- $\forall c'_i, t, a_1, a_2, \ldots, a_k, c_1, c_2, \ldots, c_k$ *subClass_of*$(a'_{ix}, a_1) \wedge$ *subClass_of*$(a'_{ix}, a_2) \wedge \ldots \wedge$ *subClass_of*$(a'_{ix}, a_k) \supset$ *cpa_Class* $(c'_1, a'_{ix}, t) \equiv$ *cpa* $(a_1, c_1, t) \wedge$ *cpa* $(a_2, c_2, t) \wedge$ *cpa* $(a_3, c_3, t) \wedge \ldots \wedge$ *cpa* $(a_k, c_k, t) \wedge [c'_i = c_1 + c_2 + c_3 + \ldots + c_k]$

From an ABC perspective, the activity_spec is the cost object of spec_act. Hence, *The cost point of an activity specification (activity-spec) is the cost point of the specifying activity (spec_act) that produces it.* Stated as an axiom in FOL:-

$$\textit{Axiom: } \forall\, (spec\_act, c, t),\ cpa(spec\_act, c, t) \wedge produces(spec\_act, activity\_spec) \supset \exists\, (activity\_spec, c', t),\ cpo(activity\_spec, c', t) \wedge (c = c') \quad \text{(FOL 79)}$$

It should be noted that the cost of specifying an activity may be a one time occurrence that may have to be "written off" over a time period during which several instances of the specified activity are executed. For example, in the pharmaceutical sector, the cost of the activity specification (usually "the patent") for the production of an antibiotic drug may be several million dollars. This cost is recovered or "written off" over a period of time, say 20 years, for which the patent is proprietary material. During this period, the cost of the activity specification is implied to be divided amongst the activity instances of the drug production as there is the notion of the activity specification being required each instance the drug is produced according to its specification. More specifically in TOVE, the activity specification of an activity is a required "resource" for the execution of each instance of the specified activity. Therefore, from a cost accuracy standpoint, it may be stated as an axiom that:-

Cost of an (executed) activity instance = [(i):- sum of costs of resources required]
+ [(ii)*:- the apportioned cost of specifying the activity]

Much of the ABC developments have focused on component (i) and have not given explicit recognition and consideration to (ii)*. Hence, the apportioned cost of specifying the activity is not adequately accounted for. Within the development of this research, we do have the means of deducing component (ii)* based upon the possible two scenarios that may present themselves in a company:-

{We denote component (ii)* as being the cost point of the resource activity_spec is denoted as the cpr(a, c, activity_spec, t) from our Cost Ontology}.

Scenario 1: A company has an estimate of the number (say n) of instances the specified activity is to be executed during some specified "write off" period of time. In this case, the apportioned cost of specifying the activity would be the cost of the specification activity divided by the n. Stated as an axiom:-

The apportioned cost of the activity specification (cpr of activity_spec) for an activity, say a1, is equivalent to the cost of the specifying activity (cpa of spec_act) divided by the number of instances (say n) estimated to occur over the desired period of time for which the cost of the specifying activity must be recovered. Stated in FOL:-

Axiom: $\forall$ ***(a, n, spec_act, activity_spec, c, t), cpa(spec_act, c, t)*** $\wedge$ ***n*** $\wedge$
produces(spec_act, activity_spec) $\wedge$ ***requires(a, activity_spec)*** $\supset$
$\exists$ ***c', cpr(a, c', activity_spec, t)*** $\wedge$ ***[c' = c / n]*** (FOL 80)

Scenario 2: A company wishes to recover its cost of specifying an activity during some period of time, say p_time. Assume that estimated or actual total activity time for all instances of the specified activity during p_time is denoted tot_act_time. We also note that based upon the Resource Ontology [Fadel et al. 94] we have the start and end time points of resource "usage" for the resource (activity_spec) through the predicate use_spec (activity_spec, a, t1, t2, q, rate, unit), where t1 and t2 are the start and end time points during which the activity_spec is used.

What is the apportioned cost of specifying the activity for a particular activity instance of duration (t2 -t1) time units? It would be based upon a *unitized time cost = [(cost of spec_act) / p_time)] x [p_time / tot_act_time]*. Then the apportioned cost of specifying activity allocated to the particular activity instance of duration (t2 - t1) would be (unitized time cost) x (t2 - t1).

We axiomatize these intuitions in FOL as follows:-

Axiom: $\forall$ ***(a, spec_act, activity_spec, c, t, t1, t2, p_time), cpa(spec_act, c, t)*** $\wedge$
produces(spec_act, activity_spec) $\wedge$ ***use_spec ((activity_spec, a, t1, t2, q, rate, unit)***
$\wedge$ ***p_time*** $\supset$
cpr(a, c', activity_spec, t) $\wedge$ ***[c' = (c/p_time) . (p_time/tot_act_time). (t2-t1)*** (FOL 81)

In closing this section, we have demonstrated that through the axioms, the cost of specifying an activity can be allocated to each activity instance. Note that a cost of specifying an activity has been obtained from the activity envelopes formed by Activity Probing and Resource Probing.

5.5.4 Differentiating between Activity Probing and Resource Probing

What, if any, are the differences between Activity Probing and Resource Probing? Activity Probing provides the means towards tracing activities that contribute to the aggregate specification activity that specifies a certain activity. On the other hand, Resource Probing provides the means of probing resources required by an activity until a frontier activity with its external resources is encountered. Resource Probing traces the activities that contribute to the cost of the resource.

Activity Probing and Resource Probing are complementary concepts put forth in this research towards tracing or building an enterprise activity infrastructure that facilitates ABC. Activity Probing requires Resource Probing. Resource Probing is complemented by Activity Probing in order to investigate costs of specifying activities. This research uses Resource Probing and Activity Probing as means towards building or tracing an enterprise activity model infrastructure based upon explicitly represented activity clusters that enable us to reason about costs using the ABC Principle.

5.6 A Formalization of a Theory of Resource Cost Units

Thus far, the discussions in this chapter have introduced intuitions, concepts and terminologies as a framework in our quest towards a theory of resource cost units. This section presents a formalization of the framework in first order logic (FOL) so that components of the theory are precise and unambiguous. Components for a Formalized Theory of Resource Cost Units are:-

- Assumptions
- Taxonomy and Definition of Terminologies

- Principle of Resource Probe
- Commit Envelope of an Internal Resource
- Reenable Envelope of a Resource
- Formalization of Overheads
- Defining a Enveloped Activity Based Enterprise Model (EABEM)
- Micro-theory to deduce Resource Cost Units.

5.6.1 Definition of Terminologies and Taxonomies

Our discussions towards the Theory of Resource Cost Units have revolved around several intuitions about the resources and activities represented in an enterprise based upon the core TOVE ontologies for activities, states, and resources. As stated, it is also imperative that all aspects to do with the conventional notions of overhead such as depreciation, supplementary benefits to employees, insurance, property taxes, etc. should be accounted for in a consistent and reliable manner if we are to achieve a fairly high level of accuracy, say plus/minus 10 percent, in our estimates for the resource cost units.

Therefore, in this section we present a Resource Taxonomy and an Activity Taxonomy from a cost perspective based upon intuitions presented in this chapter. These taxonomies maybe considered as extensions of the Resource and Activity Ontologies in TOVE as they have been founded upon the Core Ontologies in TOVE for activity, state, resource and time. As such, these extended taxonomies form an integral part of the Theory of Resource Cost Units in our formalization of ABC for enterprise modelling. Specifically, through these resource and activity taxonomies, it is our intent to represent all "entities" that are associated with contributing towards traditional overhead costs; so that, in our quest for resource cost units, our formalization of ABC maintains a consistent, accurate and reliable knowledge representation and accounting of overheads.

5.6.2 Terminologies for Activities, Resources and Cost Fluents

Companies that typically use traditional cost accounting for costing a service or product categorize costs in ways dependent upon the nature of their business. For example, manufacturing companies may have three major categories of costs. These are usually manufacturing costs made up of direct material, direct labour, and manufacturing overhead; second, selling expenses that include advertising, commissions, salaries of marketing personnel; and third, general administrative expenses which include senior executive and staff salaries, office expenses, and all other expenses not directly related to production [Rotch et al. 91].

In TOVE, it should be noted that the Core Ontologies of activities and states [Fox et al.93] and resources [Fadel & Fox, 94] that are used to represent activity clusters draw no distinctions between direct and indirect activities, direct and indirect labour, and direct and indirect materials. Therefore, the Cost Ontology for TOVE as presented in this research draws no distinctions between direct and indirect activities; and consequently, no distinction between direct and indirect costs. Our cost perspective focuses upon an activity in any situation represented as an activity cluster, that is, the activity with its enabling states and associated required resources, together with the caused states of the activity and the associated (potential) resources (or cost objects) produced.

However, from a cost perspective, our taxonomies for resources and activities must concentrate on two fronts:-

1. limit the proliferation of representing every "minute or *insignificant* activity" of an enterprise, together with representing every "minute or *insignificant* resource" required by an activity within the context of our activity cluster representation in TOVE (recall that this would be in keeping with CAM-I's CMS framework referred to earlier in Sec.5.3.2);
2. the representation of the more "nebulous entities" of traditional overhead costs associated with depreciation of factory/office buildings and equipment, taxes on real estate, rent, insurance on factory building and equipment, supplementary

employee benefits for management and unionized personnel, salaried and non-salaried personnel, and similar expenses that are incurred by enterprises.

Significant Resource: As mentioned in Sec.5.3.1, intuitively, a significant resource is one that has a high imputed cost to attain and/or has an appreciable cost impact on the cost of an activity that requires the resource. We formalize this concept of a significant resource through the usage of a nonactivity cost fluent denoted **unit_cost_spec (c,r)**, and defined as a *class* of nonactivity cost fluent where c is nonactivity cost specified of resource r. [In practice, the unit cost specified by an enterprise in respect to a significant resource may be based upon some arbitrary cost management and control policy or upon some past historical experience with the particular resource. For example, all resources that have a book value of $10,000 or more may be considered significant resources for a particular enterprise].

Axiom: $(\forall\, r, s), \exists\, c$, ***holds(unit_cost_spec (c, r), s)*** (FOL 82)

Axiom: $(\forall\, c, r)$, ***nonactivity_cost(unit_cost_spec (c, r))*** (FOL 83)

Definition: A significant resource is a resource whose specified nonactivity-based unit_cost is greater than or equal to a cost level x specified by a company

A formalized definition for significant resource in FOL is as follows:

Axiom: $\forall(r, c, x, s)$, ***holds (unit_cost_spec (c, r), s)*** $\wedge\, (c \geq x) \supset$ ***significant (r)*** (FOL 84)

Significant Activity:

A significant activity is an internal or frontier activity that requires at least one significant resource. This definition is axiomatized in FOL as follows:-

Axiom: $\forall a$, ***[internal_act (a)*** $\vee$ ***frontier_act (a)]*** $\wedge\, [\,\exists\,(r)\,]$, ***requires (a, r)*** $\wedge$ ***significant_res (r)]*** $\supset$ ***significant_act (a)*** (FOL 85)

Alternatively, *if the cost of specifying an activity is greater than or equal to a cost level y specified by the company, that activity is considered significant.*

Axiom: $\forall a$, *[internal_act (a)* $\vee$ *frontier_act (a)]* $\wedge$ *[* $\exists$ *(spec_act, c, t, y), specifies (spec_act, a)]* $\wedge$ *cpa(spec_act, c, t)* $\wedge$ *[c* $\geq$ *y]* $\supset$ *significant_act(a)* (FOL 86)

Period Cost External Resource:

Defined nonactivity cost fluents f associated with period cost external resources are based upon "traditional time period related overhead cost categories" such as building depreciation, equipment depreciation, property taxes, borrowed capital interest, insurance, salaries, wages, management supplemental benefits, and union supplemental benefits. The list of nonactivity cost fluents f associated with the period overhead entities are presented in Table 2.

[Note that employees' salaries and wages are considered as Period Cost External Resources as the physical presence of an employee for an activity requirement is viewed as a discrete and re-usable resource per see, but the "wages" or "salaries" expended on the job are consumed as resources that have an economic value expressed in terms of $ per unit time. In this research, an activity that requires a human operator has two conjunctive enabling states - one that uses the operator as a re-usable, discrete resource; and the second that consumes the operator_wages/salaries as a consumable, rated resource].

TABLE 2 Defined Nonactivity_cost Fluents Associated to Commonly Identified Traditional Time Period Related Overhead Cost Categories

Period Overhead Cost Categories	*Nonactivity_cost Fluents*
building depreciation	***bldg_depr_cost (tps, tc_ext, tt_act, r)***
equipment depreciation	***equip_depr_cost (tps, tc_ext, tt_act, r)***
property taxes	***property_tax_cost (tps, tc_ext, tt_act, r)***
borrowed capital interest	***borr_capital_cost (tps, tc_ext, tt_act, r)***
insurance	***insurance_cost (tps, tc_ext, tt_act, r)***
salaries	***salary_cost (tps, tc_ext, tt_act, r)***
wages	***wage_cost (tps, tc_ext, tt_act, r)***
management supplemental benefits	***mgt_benefit_cost (tps, tc_ext, tt_act, r)***
union supplemental benefits	***union_benefit_cost (tps, tc_ext, tt_act, r)***
leases	***lease_cost (tps, tc_ext, tt_act, r)***

The list of traditional time period overhead cost categories in Table 2 is not exhaustive, but is "guesstimated" to cover about 95 percent of time period overhead cost categories for most enterprises. If an enterprise has other overhead cost entities it wishes to include in Table 2 (e.g., training, safety, hiring and health) a corresponding class of nonactivity cost fluent of the form *f(tps, tc_ext, tt_act, r)* may be defined where:-

- *f* is a predicate denoting the class of nonactivity_cost fluent for the overhead cost,
- *tps* denotes the time period under study, (e.g., 1 year, 6 months, 1 quarter, etc.)
- *tc_ext* is an externally given total nonactivity-based cost applicable to tps, (e.g., if the fixed overhead of depreciation is under study for tps = 1year, then tc_ext would be the annual depreciation cost),
- *tt_act* is a total actual time or total estimated time for the number of instances that have occurred or are expected to occur in tps,
- *r* is the name of the external resource associated with the nonactivity cost fluent.

Each of the cost fluents of Table 2 is defined as a class of nonactivity_cost fluent. The axioms for each of these definitions are:-

Axiom: ∀ (r, tps, tc_ext, tt_act), nonactivity_cost(bldg_depr_cost (tps, tc_ext, tt_act, r)) **(FOL 87)**

Axiom: ∀ (r, tps, tc_ext, tt_act), nonactivity_cost(equip_depr_cost (tps, tc_ext, tt_act, r)) **(FOL 88)**

Axiom: ∀ (r, tps, tc_ext, tt_act), nonactivity_cost(property_tax (tps, tc_ext, tt_act, r)) **(FOL 89)**

Axiom : ∀ (r, tps, tc_ext, tt_act), nonactivity_cost(borr_capital_cost (tps, tc_ext, tt_act, r)) **(FOL 90)**

Axiom: ∀ (r, tps, tc_ext, tt_act), nonactivity_cost(insurance_cost (tps, tc_ext, tt_act, r)) **(FOL 91)**

Axiom: ∀ (r, tps, tc_ext, tt_act), nonactivity_cost(salary_cost (tps, tc_ext, tt_act, r)) **(FOL 92)**

Axiom: $\forall$ ***(r, tps, tc_ext, tt_act), nonactivity_cost(wage_cost (tps, tc_ext, tt_act, r))*** (FOL 93)

Axiom: $\forall$ ***(r, tps, tc_ext, tt_act), nonactivity_cost(mgt__benefit_cost(tps, tc_ext, tt_act, r))*** (FOL 94)

Axiom: $\forall$ ***(r, tps, tc_ext, tt_act),nonactivity_cost(union_benefit_cost(tps, tc_ext, tt_act, r))*** (FOL 95)

Axiom: $\forall$ ***(r, tps, tc_ext, tt_act), nonactivity_cost(lease_cost(tps, tc_ext, tt_act, r))*** (FOL 96)

The definitions for the nonactivity cost fluents in Axioms [(FOL 87) - (FOL 96)] are used in formalizing a definition for Period Cost External Resource as follows:-

> *Definition: For all situations s, Period Cost External Resource is a class of external resource for which there exists a nonactivity_cost fluent f associated with the resource such that f represents one of the following cost related categories - building depreciation, equipment depreciation, property tax, borrowed capital interest, insurance cost, salaries, wages, management supplemental benefits, union supplemental benefits, and leases; and denoted as* ***bldg_depr_cost (tps, tc_ext, tt_act, r)****,* ***equip_depr_cost (tps, tc_ext, tt_act, r)****,* ***property_tax (tps, tc_ext, tt_act, r)****,* ***borr_capital_cost (tps, tc_ext, tt_act, r)****,* ***insurance_cost (tps, tc_ext, tt_act, r)****,* ***salary_cost (tps, tc_ext, tt_act, r)****,* ***wage_cost (tps, tc_ext, tt_act, r)****,* ***mgt_benefit_cost (tps, tc_ext, tt_act, r)****,* ***union_benefit_cost (tps, tc_ext, tt_act, r)****,* ***and lease_cost (tps, tc_ext, tt_act, r)*** *respectively, where tps denotes time period under study, tc_ext denotes a nonactivity-based total cost to be distributed over time period of study for the particular overhead cost category, and tt_act is actual total time or total estimated time for instances of activities requiring the particular cost category.*

We axiomatize the definition of Period Cost External Resource in FOL as follows:-

$$\textit{Axiom: } \forall (s, r), external\_res(r) \wedge [\ (\exists f, tps, tc\_ext, tt\_act), nonactivity\_cost(f) \wedge holds(f, s) \wedge [(f = bldg\_depr\_cost\ (tps, tc\_ext, tt\_act, r)) \vee (f = equip\_depr\_cost\ (tps, tc\_ext, tt\_act, r)) \vee (f = property\_tax\ (tps, tc\_ext, tt\_act, r)) \vee (f = borr\_capital\_cost\ (tps, tc\_ext, tt\_act, r)) \vee (f = insurance\_cost\ (tps, tc\_ext, tt\_act, r)) \vee (f = salary\_cost\ (tps, tc\_ext, tt\_act, r)) \vee (f = wage\_cost\ (tps, tc\_ext, tt\_act, r)) \vee (f = mgt\_benefit\_cost\ (tps, tc\_ext, tt\_act, r)) \vee (f = union\_benefit\_cost\ (tps, tc\_ext, tt\_act, r)) \vee (f = lease\_cost\ (tps, tc\_ext, tt\_act, r))] \supset period\_cost\_ext\_res(r) \qquad \text{(FOL 97)}$$

Based upon the definition of Period Cost External Resource, we state the following axiom that enables us to deduce the unit cost of a resource that belongs to the class of Period Cost External Resource:-

> *For all situations s, the unit cost of a period cost external resource is equivalent to a nonactivity-based total cost, denoted tc_ext, divided by a total time period, denoted tt_act; where tc_ext and tt_act are specified for the nonactivity cost fluent f associated with the resource.*

$$\textit{Axiom*: } \forall(s, r), period\_cost\_ext\_res(r) \supset \neg\ (\exists a), produces(a, r) \wedge (\exists f, tps, tc\_ext, tt\_act), nonactivity\_cost(f) \wedge holds(f, s) \wedge [(f = bldg\_depr\_cost\ (tps, tc\_ext, tt\_act, r)) \vee (f = equip\_depr\_cost\ (tps, tc\_ext, tt\_act, r)) \vee (f = property\_tax\ (tps, tc\_ext, tt\_act, r)) \vee (f = borr\_capital\_cost\ (tps, tc\_ext, tt\_act, r)) \vee (f = insurance\_cost\ (tps, tc\_ext, tt\_act, r)) \vee (f = salary\_cost\ (tps, tc\_ext, tt\_act, r)) \vee (f = wage\_cost\ (tps, tc\_ext, tt\_act, r)) \vee (f = mgt\_\_benefit\_cost\ (tps, tc\_ext, tt\_act, r)) \vee (f = union\_benefit\_cost\ (tps, tc\_ext, tt\_act, r)) \vee (f = lease\_cost\ (tps, tc\_ext, tt\_act, r))] \supset (\exists c), holds(unit\_cost\_ext\ (c, r), s) \wedge \{[\ c = (tc\_ext / tps) . (tps / tt\_act)] \vee [c = tc\_ext / tt\_act]\} \qquad \text{(FOL 98)}$$

[* Note: Fixed annual overhead costs such as depreciation, insurance, leases, benefits, etc. get distributed based upon actual activity utilization. In other words, lower the actual utilization of a plant, higher will be these overhead components in the cost of products; and vice versa.

For example, suppose factory building depreciation cost per annum is $75,000 and the actual building usage is 750 hours for the year. We are interested in deducing the unit cost of depreciation for the year in which the actual building usage is 750 hours.Then tps = 1 year, tt_act = 750 hours, tc_ext = $75,000 per year. According to Axiom (FOL 98), the unit cost of depreciation would be {c = (tc_ext / tps) (tps / tt_act) = ($75,000 / 1 year) (1 year / 750 hours) } = $100 per hour. If the actual utilization of the building were increased to 1, 000 hours, the unit cost of depreciation would now decrease to $75 per hour. In other words, higher resource utilization leads to lower product cost; or high idle resource capacity, leads to high product costs.]

Non-Period Cost External Resource:

Table 3 lists defined nonactivity cost fluents f associated with non-period cost external resources that are based upon "traditional non-period overhead cost categories" such as material costs and utility costs pertaining to hydro, heat and water. Each of the cost fluents of Table 3 is defined as a class of nonactivity cost fluent. The axioms for each of these definitions are:-

Axiom: (∀ ***r, tc_ext, tqty_ext), nonactivity_cost(heat_cost (tc_ext, tqty_ext, r))*** (FOL 99)

Axiom: (∀***r, tc_ext, tqty_ext), nonactivity_cost(hydro_cost (tc_ext, tqty_ext, r))*** (FOL 100)

Axiom:(∀***r, tc_ext, tqty_ext), nonactivity_cost(water_cost (tc_ext, tqty_ext, r))***(FOL 101)

Axiom: (∀***r, tc_ext, tqty_ext), nonactivity_cost(matl_cost (tc_ext, tqty_ext, r))***(FOL 102)

TABLE 3 Defined Non-Period Nonactivity_cost Fluents Associated to Commonly Identified Traditional Non-Period Related Overhead Cost Categories

Non-Period Overhead Cost Categories	*Non-Period Nonactivity_cost Fluents*
hydro	***hydro_cost (tc_ext, tqty_ext, r)***
heat	***heat_cost (tc_ext, tqty_ext, r)***
water	***water_cost (tc_ext, tqty_ext, r)***
"indirect materials"	***matl_cost (tc_ext, tqty_ext, r)***

The definitions for the nonactivity cost fluents in Axioms (FOL 99) - (FOL 102) are used in the FOL definition for the class of non-period cost external resources as follows:-

Definition: For all situations s, Non-Period Cost External Resource is a class of external resource for which there exists a nonactivity_cost fluent f associated with the resource such that f represents one of the following cost related categories - hydro, heat, water, and "indirect materials"; denoted as ***hydro_cost (tc_ext, tqty_ext, r)***, ***heat_cost (tc_ext, tqty_ext, r)***, ***water_cost (tc_ext, tqty_ext, r)*** *and* ***matl_cost (tc_ext, tqty_ext, r)*** *respectively; where tc_ext denotes a specified nonactivity-based total cost parameter distributed over the total quantity parameter tqty_ext associated with the particular cost category.*

We axiomatize the definition of Non-Period Cost External Resource in FOL as follows:-

$$\textit{Axiom}: \forall (s, r),\ \textit{external\_res}(r) \wedge [\ (\exists f, tc\_ext, tqty\_ext),\ \textit{nonactivity\_cost}(f) \wedge \textit{holds}(f, s) \wedge [(f = \textit{hydro\_cost}\ (tc\_ext, tqty\_ext, r)) \vee (f = \textit{heat\_cost}\ (tc\_ext, tqty\_ext, r)) \vee (f = \textit{water\_cost}\ (tc\_ext, tqty\_ext, r)) \vee (f = \textit{matl\_cost}\ (tc\_ext, tqty\_ext, r))] \supset \textit{non\_period\_cost\_ext\_res}(r) \qquad \text{(FOL 103)}$$

Using Axiom (FOL 105), the following axiom is stated to deduce the unit cost of a non-period cost external resource:-

For all situations s, the unit cost of a non-period cost external resource is equivalent to a nonactivity-based total cost, denoted tc_ext, divided by a total quantity, denoted tqty_ext; where tc_ext and tqty_ext are specified parameters for the nonactivity cost fluent f associated with the resource.

Axiom: $\forall(s, r)$, **non_period_cost_ext_res(r)** $\supset \neg (\exists a)$, **produces(a, r)** $\wedge \exists$ **(f, tc_ext, tqty_ext), nonactivity_cost(f)** $\wedge$ **holds(f, s)** $\wedge$ **[(f = hydro_cost (tc_ext, tqty_ext, r))** $\vee$ **(f = heat_cost (tc_ext, tqty_ext, r))** $\vee$ **(f = water_cost (tc_ext, tqty_ext, r))** $\vee$ **(f = matl_cost (tc_ext, tqty_ext, r))]** $\supset (\exists c)$, **holds(unit_cost_ext(c, r), s))** $\wedge$ **[c = tc_ext / tqty_ext]** (FOL 104)

Non-Period Cost Internal Resource:

By definition, an internal resource is produced by an enterprise activity. From an ABC perspective, the resource produced is the cost object of the activity that produces it. In terms of the Cost Ontology for TOVE in Ch. 4, the relationship between the producing activity and the resource produced is defined at all times t with the predicate has_cost_order (x, a, t), where x = cost object, and a = activity that produces the object (in this case, the resource). The inverse of the has_cost_order relation is the relation cost_order_of.

If an activity a produces resource r at time t, then r is the cost order of a at time t.

Axiom: $\forall$ **(t, a, r), holdsT(produces(a, r), t)** $\supset$ **cost_order_of(r, a, t)** (FOL 105)

In Ch. 4, the cost point of an activity (cpa) and the cost point of an order (cpo) have been defined in Axiom (FOL 34) and Axiom (FOL 36) respectively. From a cost perspective, assuming that n different cost orders are simultaneously produced by an activity instance, then the cost of that activity must be distributed amongst those n cost orders.

In order to formalize the equal distribution of activity costs amongst different cost orders produced at the same time, the following axiom schema in FOL is stated:-

The cost point of an activity at time t is equally distributed amongst its n different cost orders at time t in order to compute the cost point of each cost order.

[Note: Axiom (FOL 34) of Ch. 4 defines the cost point of an activity as cpa(a, c, t).]

$$\textbf{Axiom (Schema): } \forall (a, r_i, t, n, c), cpa(a, c, t) \wedge [cost\_order\_of(r_1, a, t) \wedge cost\_order\_of(r_2, a, t) \wedge cost\_order\_of(r_3, a, t) \wedge .... \wedge cost\_order\_of(r_n, a, t)] \supset (\exists c'), cpo(c', r_i, t) \wedge [c' = c / n] \wedge (i = 1, 2, 3, ..., n) \quad \text{(FOL 106)}$$

Definition: A non-period cost internal resource r is a class of internal resource for which there exist an activity a that produces it in quantity q as specified in the production specification of the resource.

[* Note: The Resource Ontology [Fadel et al. 94] in TOVE defines the production specification of a resource with the term produce_spec (r, a, st, t, q, rate, unit) where r = resource, a = activity, st = state, t = time point, q = quantity produced, rate = rate of production, unit = unit of measure. According to the Resource Ontology, if q = rate, the resource is produced as a discrete quantity batch. However, if q = rate, the resource is produced at some continuous rate which is equal to the rate parameter. In other words, from the produce_spec of the Resource Ontology, we may deduce if a resource is produced as a discrete batch quantity, or the total quantity produced based upon the continuous rate of production].

The definition of non-period cost internal resource is axiomatized in FOL as follows:-

$$\textbf{Axiom: } \forall (r), internal\_res(r) \supset (\exists a, st, t, q, rate, unit), produces(a, r) \wedge produce\_spec(r, a, st, t, q, rate, unit) \supset non\_period\_cost\_int\_res(r) \quad \text{(FOL 107)}$$

From Axiom (FOL 103), we are able to deduce the cost point order of a resource r, denoted as cpo (c, r, t) where c = cost of resource produced at time point t. We also know the quantity q of resource produced from the produce_spec as used in Axiom (FOL 104). Hence, in this case, the cost of unit resource is equivalent to the cost of resource produced divided by the quantity.

To enable us to deduce the unit cost of a non-period cost internal resource, we state the following axiom:-

> *The unit cost of a non-period cost internal resource is equivalent to the cost point order of the resource produced divided by the resource quantity produced as given by the produce specifications for the resource.*

This is formalized in FOL as follows:-

$$(\forall\, r),\ non\_period\_cost\_int\_res(r) \supset \exists\, (a, st, c', t, q, rate, unit), produces(a, r) \wedge cpo(c', r, t)] \wedge produce\_spec(r, a, st, t, q, rate, unit) \supset [\ (\exists\, c), holdsT(unit\_cost\_int(c, r), t) \wedge (c = c' / q)] \quad \text{(FOL 108)}$$

[Example, suppose at time t, the cost point of an activity instance, named bolt_washer_mfg, is $100. Assume that at time t, the activity simultaneously produces a batch of 5 bolts and a batch of 10 washers, where the bolts and washers are resources to some other subsequent activity. The batch of 5 bolts and the batch of 10 washers are *2 cost orders* of the bolt_washer_mfg activity. First, from an ABC perspective, the cost point of each cost order is *$50 ($100/2 cost orders)* according to Axiom (FOL 106). Secondly, the unit cost of each bolt is *$10 ($50/5 bolts)*, whereas the unit cost of each washer is *$5 ($50/ 10 washers)* according to Axiom (FOL 108)].

<u>**Period Cost Internal Resource:**</u>

A hypothetical example for the concept of period cost internal resource is offered here. Assume an activity called contract_negotiating_activity. Assume that the activity-based cost of this activity is $15,000 based upon the definition of cpa(a, c, t). Assume this activity causes a maintenance_service_contract to be produced as its only cost order. Then the cost point order of the resource, i.e., maintenance_service_contract, is $15,000 = ($15,000/1). Assume that the maintenance_service_contract provides 100 hours of maintenance service. We may consider the maintenance_service_contract for 100 hours as an internal resource. In this case, we consider c' = $15,000 and an associated time quantity to be the 100 hours of maintenance service to be provided by the contract. Then, intuitively, we may deduce that a contributory unit cost value per hour of maintenance service usage is c = $15,000/ 100 hours = $150 / hour. Note that the unit cost of $150/hour can only be arrived *if and only if* we are given "total time associated with internal resource" as being 100 hours with certainty*.

[* Note: If more than 100 hours of maintenance are used, presumably the activity cost of contract_negotiating_activity would increase as more "negotiations" may have to be done. This means that the former $150 / hour rate would be changed depending upon the extra hours negotiated and the activity cost of the second instance of contract_negotiating_activity. In such a case, a more accurate product cost can only be obtained on through a *forensic analysis or retrospective basis by using a metric other than $150 / hour.* On the other hand, it is recommended that, if the company is uncertain about more, less or no hours of maintenance being used, the company may treat "maintenance" as a period overhead cost category to be added to Table 2. This would ensure that "maintenance" is treated like a "lease".]

We formalize these intuitions with regard to period cost internal resource as follows:-

Definition: Period Cost Internal Resource is a class of resource produced by an activity, and has a time quantity, denoted tm_qty, associated with it.

Axiom : $\forall$ ***(a, r),*** $\exists$ ***tm_qty, produce (a, r)*** $\wedge$ ***tm_qty*** $\supset$ ***period_cost_int_res(r)*** (FOL 109)

Using Axiom (FOL 109), the following axiom is stated to deduce the unit cost of a period cost internal resource:-

The unit cost c of a period cost internal resource r is equivalent to the cost point order of r divided by its associated specified time quantity denoted by tm_qty.

This is formalized in FOL as follows:-

Axiom: ($\forall$ ***r), period_cost_int_res(r)*** $\supset \exists$ ***(a, c', tm_qty, t), produces(a, r)*** $\wedge$ ***cpo(c', r, t)]*** $\wedge$
tm_qty $\supset$ ***[(*** $\exists$ ***c), holdsT(unit_cost_int(c, r), t)*** $\wedge$ ***(c = c' / tm_qty)*** (FOL 110)

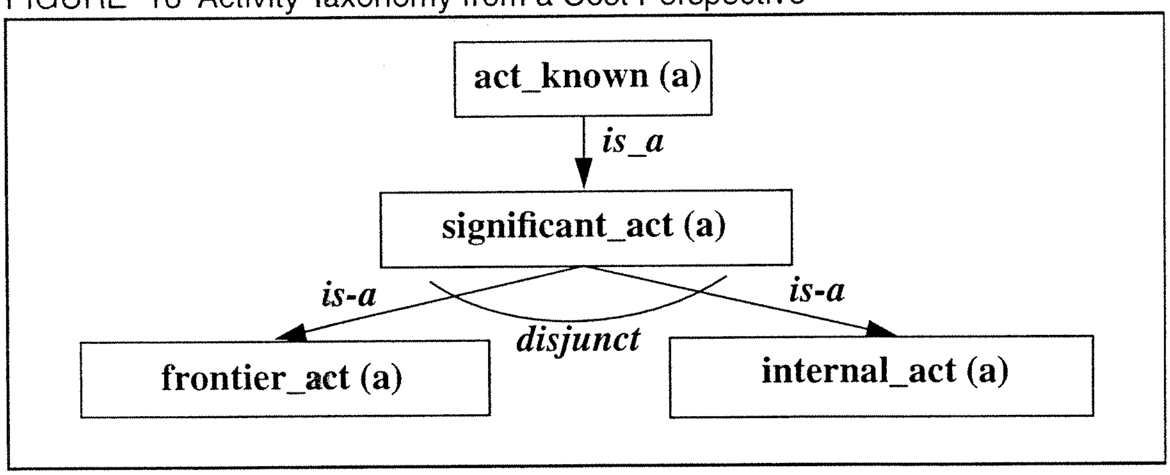
In closing this section, we note the following:-

1. unit cost of Period Cost External Resource is deduced with Axiom (FOL 98);
2. unit cost of Non-Period Cost External Resource is deduced with Axiom (FOL 104);
3. unit cost of Non-Period Cost Internal Resource is deduced with Axiom (FOL 108);
4. unit cost of Period Cost Internal Resource is deduced with Axiom (FOL 110).

5.7 Taxonomies: Activities, Resources & Cost Fluents

The previous section explained the semantics of terms for an enterprise model. This section presents taxonomies for activities, resources and cost fluents from a *cost perspective*, wherein a taxonomy is a tree structure displaying is-a relationships [Brachman 85] between classes in a hierarchical manner.

FIGURE 16 Activity Taxonomy from a Cost Perspective

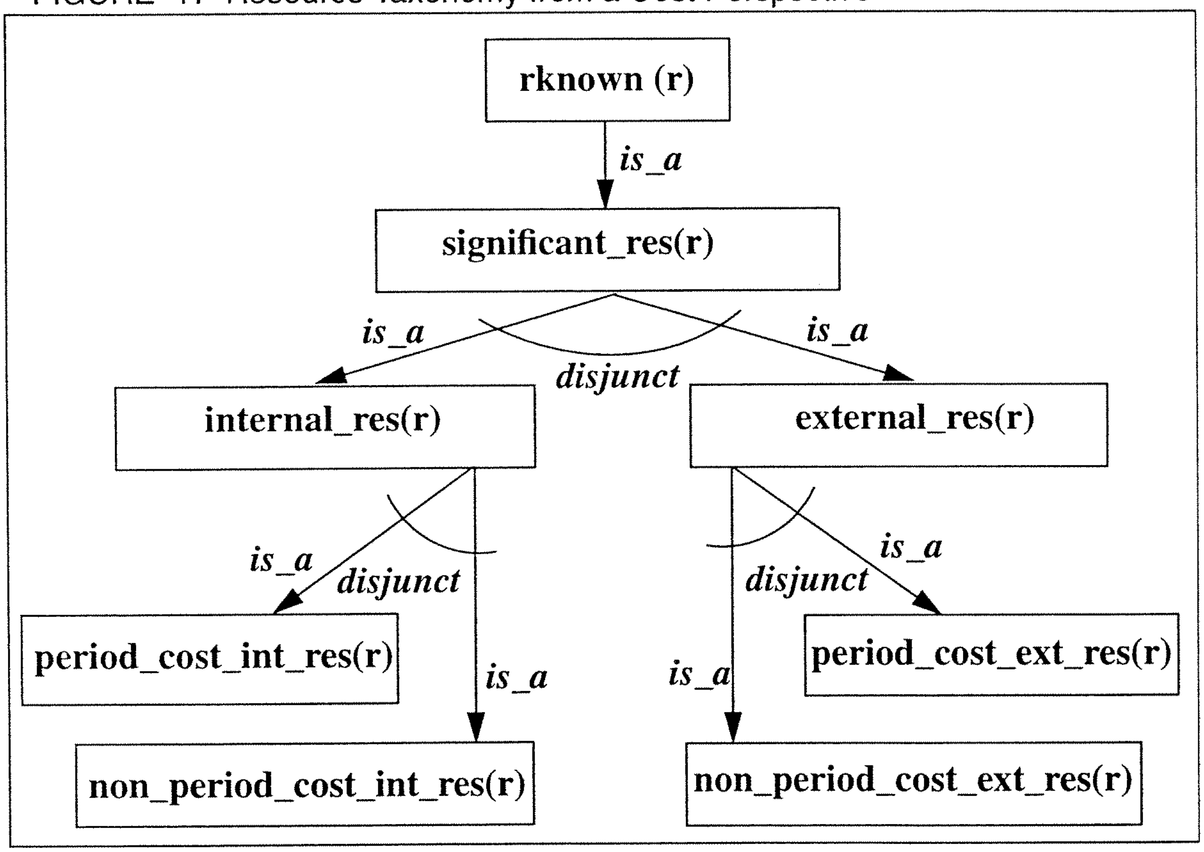


Since we wish to reason about costs through a populated enterprise model infrastructure based upon activities and resources, our starting points for the Activity Taxonomy (Fig. 16) and Resource Taxonomy (Fig. 17) are the classes of known activities, act_known (a), and known resources, rknown (r), respectively. For purposes of limiting the size of the populated model and for economic expediency, the activity and resource taxonomies focus upon significant activities and significant resources respectively.

A significant activity is a frontier activity or an internal activity as shown in Fig. 16.

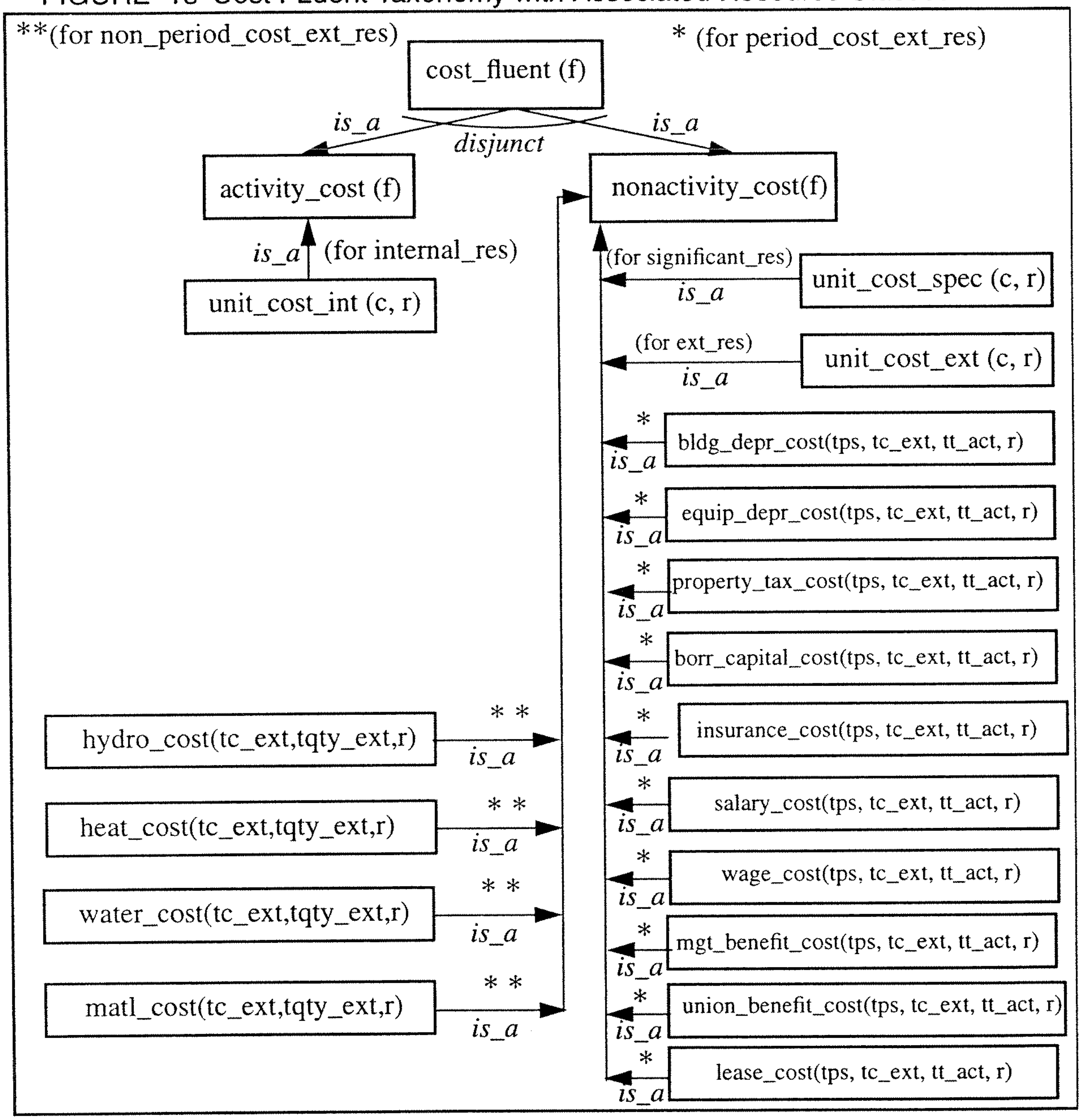
The taxonomy in Fig. 17 indicates that a significant resource is an internal resource or an external resource. An internal resource may be a period cost internal resource or a non-period cost internal resource. Similarly, an external resource falls into the class of period cost external resources or the class of non-period cost external resources.

FIGURE 17 Resource Taxonomy from a Cost Perspective



The Cost Fluent Taxonomy (Fig. 18) serves two purposes. First, the class of cost fluents is segregated into activity-based cost fluents and nonactivity-based cost fluents so that unit costs of resources may be deduced based upon some period of time or some quantity.

FIGURE 18 Cost FLuent Taxonomy with Associated Resource Classes



associated with a resource, thereby drawing upon the distinction between the class of period cost resources and non-period cost resources respectively. Second, the Cost Fluent Taxonomy contains several defined classes of nonactivity-based cost fluents (refer Tables 2 and 3) with a view to *formalizing the traditional concepts of overhead costs* associated with building/equipment depreciation, property taxes, borrowed capital interest, insurance, salaries, wages, management/union supplemental benefits, leases, hydro, heat, water and materials. The defined classes of nonactivity-based cost fluents enable us to include in a consistent, accurate and unambiguous manner the nebulous entities of overhead costs in our deductions of the unit costs of resources. Further, the nonactivity cost fluents complement the Resource Taxonomy "to reason with and make visual" the nebulous entities of overheads within the context of the activity cluster of an activity model in TOVE.

5.8 Enveloped Activity Based Enterprise Model (EABEM)

Throughout this research, we have used the activity cluster as being the nucleus in building an enterprise model. Central to doing this and as stated by [Sathi et al. 85], the representation of activity knowledge is important to any application which must reason about activities. "Activities" are central to ABC and ABM as was discussed in the *CAM-I cross* in Ch. 2. The activity representation in TOVE is similar across the various application domains, be they service, public, government, or manufacturing sector oriented. Its purpose in this research is to aid enterprises in getting solutions to their "competency questions" directed towards strategic cost management.

Resource Probing has been motivated by our quest to determine the resource cost units. Activity Probing has been motivated towards determining the cost of specifying an activity. These are conceptual procedures to direct the enterprise model builder towards investigating and tracing activities for the purposes of costing. These procedures have been founded upon the activity cluster representation in TOVE, i.e., the Core Ontology in TOVE for activity, state, and resource. From an Ontological Engineering perspective, this enables us to consistently apply "a representation language, (in our case, a Cost Ontology of Ch. 4), which satisfies the criteria of completeness, precision, and lack of ambiguity" [Sathi et al. 85]. From a costing perspective, it has enabled us to focus upon a Formalization of the ABC Principle with the goal of providing a solution to many of the identified Costing Problems (Ch. 2) faced by industries towards achieving strategic cost management. Resource Probing facilitates the trace of activities that contribute to the cost of a resource. Activity Probing promotes the trace of activities that contribute to the cost of specifying an activity. Hence, the cost oriented perspectives taken in Resource Probing and Activity Probing lend some support to this research being "applicable and practical" for the real world.

From an ABC perspective, Resource Probing facilitates the trace of activities that contribute to the cost of a resource. Resource Probing brings "closure" to the activity model by terminating the probe at a Frontier Activity - defined as requiring only External Resources. *Termination or "closure" in Resource Probing is based upon the assumption that External Resources are acquired from external suppliers at known acquisition costs that are "nonactivity-based" from the perspective of the enterprise being modelled.* Resource Probing introduced the concept of *envelopes* (Commit Envelope and Reenable Envelope) as being layers of activity clusters uncovered through the probe and that surround the probed resource. We rationalized this deepening of the probe and the envelopes upto some terminal frontier made up of Frontier Activities and External Resources. For practicality, and based upon the Principles of Cost Management Systems put forth by CAM-I (Ch. 2), we focused the resource probe and the contents of envelopes to significant activities and significant resources.

From an ABC perspective, Activity Probing was motivated towards our quest to obtain the cost of specifying an activity. It is founded upon a "specification activity" *specifying* an activity. Activity Probing facilitates the trace of subClasses of activities that compose a specification activity. In doing so, it unveils classes of activities such as research and development, engineering services, marketing, negotiations, financial planning, human resourcing, etc. - activities often sweepingly alluded to as "overhead activities" by many companies. Each of these activities require resources. These resources are then probed with Resource Probing to bring closure to activity envelopes frontiered or bounded by Frontier Activities and External Resources. Closure to Activity Probing is brought about by Resource Probing.

Hence, Resource Probing complements Activity Probing to bring closure to the latter. However, Resource Probing is independent of Activity Probing.

Therefore, Resource Probing and Activity Probing structure an activity model of an enterprise to consist of envelopes of significant activities and significant resources. It is these envelopes that provide us the activity infrastructure to reason about costs consistent to a Formalization of ABC according to the Cost Ontology of Ch. 4. Hence, the coining of the term Enveloped Activity Based Enterprise Model - acronym, EABEM.

5.8.1 A Formalized Schema for EABEM

An Enveloped Activity Based Enterprise Model (EABEM) is represented as follows:

- $E \equiv [\Sigma_{\text{internal resources}} \cap (\xi_{sig})] \cup [\Sigma_{\text{external resources}} \cap (\xi_{sig})] \cup [\Sigma_{\text{activities}} \cap (\eta_{sig})] \cup [\Sigma_{\text{frontier activities}}]$

where,

- $[\Sigma_{\text{internal resources}} \cap (\xi_{sig})]$: the set of sentences defining significant internal resources of the enterprise
- $[\Sigma_{\text{external resources}} \cap (\xi_{sig})]$: the set of sentences defining significant external resources (aka frontier resources) to the enterprise
- $[\Sigma_{\text{activities}} \cap (\eta_{sig})]$: the set of sentences defining significant activities of the enterprise
- $[\Sigma_{\text{frontier activities}}]$: the set of sentences defining the frontier activities of the enterprise

5.9 Principle of Resource Probing

The probing of resources required by an activity has been central towards the formation of activity envelopes in an enterprise model. To guide the enterprise activity builder, the Principle of Resource Probing states:-

Probing a resource R_i involves identifying some known enterprise activity whose caused state produces R_i. If the probing of R_i yields no such activity, R_i is considered an external resource, and the probing of R_i is terminated. However, if the probing of R_i reveals one such activity, say A_{1i}, then A_{1i} is a sub_activity of an aggregate activity $\varnothing_i$, termed the commit_envelope of R_i. Each resource $R_k\_A_{1i}$, $k = 1, 2, ..., m$ of A_{1i} is probed. If each $R_k\_A_{1i}$ of A_{1i} is a frontier resource, then A_{1i}

is considered a frontier activity, and $\varnothing_i$ *is said to be complete for* R_i*. If the probing of some* $R_k\_A_{1i}$ *reveals some activity, say* A_{2i_k}*, whose caused state produces* $R_k\_A_{1i}$*, then* A_{2i_k} *is considered a sub_activity of* $\varnothing_i$*.* A_{1i} *is enable_cause related to* A_{2i_k} *to form an enable_cause chain for* R_i*. The probing of resources for* A_{2i_k} *is repeated as in* A_{1i} *until a frontier activity ends an enable_cause chain in* $\varnothing_i$*. Each* $\varnothing_i$ *for* R_i *consists of sub_activities that form enable_cause chains that end in frontier activities that bring closure to the chains.*

Probing is applied to each R_i $(i=1, 2, 3, ..., n)$ *so that* $\varnothing_i$ *is complete for each* R_i*. The resultant activity model developed is referred to as an Enveloped Activity Based Enterprise Model (EABEM) that is bounded by external resources only.*

5.10 The Principle of Activity Probing

The Principle of Activity Probing states that:-

1. Each (significant) activity must have a specification that defines it. The probe of the activity establishes a link between the probed activity and its specification activity through the *specifies* relationship.
2. All significant activities are identified that are *subClasses_of* the specification activity.
3. Resources required by the subClass activities of a specification activity are probed according to the Principle of Resource Probing to bring closure to Activity Probing.

5.11 A Formalization of Overheads

According to Webster's dictionary, the term overhead is "the operating expenses of a business, including the costs of rent, utilities, interior decoration, and taxes, and excluding labor and materials". Overhead has other common synonyms used in practice such as indirect expenses, manufacturing overhead, factory burden, factory overhead, indirect burden, and factory expense. No matter what synonym is used, *the basic concept of overhead envisages all costs except direct labour and direct materials.*

TABLE 4 A Formalization of Traditional "Overhead Classes"

	FORMALIZATION		
"Overhead Classes"	*Terms*	*Axioms*	*Taxonomies*
(Time) Period Related:- building depreciation, equipment depreciation, property taxes, borrowed capital interest, insurance, salaries, wages, management supplemental benefits, union supplemental benefits, leases	***Period Cost External Resource***	***Definition: (FOL 97)***	***Resource Taxonomy: Figure 17***
	Non-Activity Cost Fluents	***Definitions: (FOL 87) - (FOL 96)***	***Cost Fluent Taxonomy: Figure 18***
Non (Time) Period Related: hydro, heat, water, "indirect materials"	***Non-Period Cost External Resource***	***Definition: (FOL 103)***	***Resource Taxonomy: Figure 17***
	Non-Activity Cost Fluents	***Definitions: (FOL 99) - (FOL 102)***	***Cost Fluent Taxonomy: Figure 18***

In this research, traditional overhead entities such as depreciation, taxes, insurance, etc., (refer Table 2) have been formalized as Period Cost External Resources defined in Axiom (FOL 97), and associated with Nonactivity_Cost Fluents defined in Axioms (FOL 87) - (FOL 96). Other overhead entities such as "indirect materials" and utilities (hydro, heat, water), (refer Table 3) are formalized as Non-Period Cost External Resources defined in Axiom (FOL 103), and associated with Nonactivity_Cost Fluents defined in Axioms (FOL 99) - (FOL 102). A Formalization of Overheads is achieved with Terms, Axioms, and Taxonomies shown in Table 4.

The formalization of overheads enables one to represent traditional overhead entities within the activity cluster representation so that costs are directed consistently from resource to activity to cost objects produced by the caused state of an activity through the formalization of cpr, cpa, and cpo of Ch. 4. Therefore, from the perspective of this research, "traditional overhead entities" are no longer "indirect costs", but are in fact "direct costs" to the cost objects of activities. This makes the impact of costs of activities on business processes more transparent for purposes of strategic cost management and business process re-engineering.

5.12 Deductive Reasoning about Costs with ABC

Our formalization of ABC in Ch. 4 led to the development of cost point of a resource (cpr), cost point of an activity (cpa), and cost point of an order (cpo). These were premised upon knowing the cost units of a resource, viz., committed resource cost unit, enabled resource cost unit, disenabled resource cost unit, reenabled resource cost unit. In this section, a Micro-Theory of Resource Cost Units is presented based upon a minimal ontological commitment [Gruber 91] - a design guideline to restrict axioms of an ontology to those required to minimally describe a domain. The logic to reason about costs with ABC is then presented based upon the Formalization for Theory of Resource Cost Units, a Micro-Theory for Resource Cost Units, together with cpr, cpa, and cpo.

5.12.1 Deducing Committed Resource Cost Unit of a Resource

From a cost perspective, a resource belongs to any one of the four resource classes:-

1. Period Cost External Resource
2. Non-Period Cost External Resource
3. Non-Period Cost Internal Resource
4. Period Cost Internal Resource

Axioms (FOL 98) and (FOL 104) applicable to Period Cost External Resource and Non-Period Cost External Resource respectively enable us to deduce the unit cost of the resource through the predicate **unit_cost_ext(c, r)**.

Axioms (FOL 108) and (FOL 110) applicable to Non-Period Cost Internal Resource and Period Cost Internal Resource respectively enable us to deduce the unit cost of the resource through the predicate **unit_cost_int(c, r)**.

The committed resource cost unit is a primitive term in Ch. 4 represented by the predicate **committed_res_cost_unit (a, r, q, v_1)**. We state the following axioms to deduce the committed resource cost unit for an external resource and an internal resource:-

Given that a resource is an external resource, the unit cost of the external resource is equivalent to its committed resource cost unit.

Axiom: $\forall(a, r, s, q, v_1)$, ***external_res(r)*** $\wedge$ ***holds (unit_cost_ext(c, r), s)*** $\supset \exists\, v_1$, ***(holds (committed_res_cost_unit (a, r, q, v_1), s)*** $\wedge$ $(v_1 = c)$ **(FOL 111)**

Given that a resource is an internal resource, the unit cost of the internal resource is equivalent to its committed resource cost unit.

Axiom: $\forall(a, r, s, q, v_1)$, ***internal_res(r)*** $\wedge$ ***holds (unit_cost_int(c, r), s)*** $\supset \exists\, v_1$, ***(holds (committed_res_cost_unit (a, r, q, v_1), s)*** $\wedge$ $(v_1 = c)$ **(FOL 112)**

5.12.2 Micro-theory of Resource Cost Units

From the perspectives of knowledge-based systems and Ontological Engineering, a micro-theory is a formal model of knowledge required to solve a problem or task in a domain or describe a subset of the domain in detail [Guha et al. 90] [Guida 94]. It is separate from, but constructed upon an ontology.

The task at hand is to deduce resource cost units of a resource. Resource cost units in our Cost Domain include committed resource cost unit, enabled resource cost unit, disenabled resource cost unit, and reenabled resource cost unit. Knowing the resource cost units of a resource and the quantities of resource used or consumed by the enabling states of an activity, one can deduce the cost point of a resource, cpr, for the activity.

The formal model of knowledge required to deduce resource cost units is separate from, but constructed upon the Cost Ontology of Ch. 4. Our formalization include primitive terms, i.e., predicates which are never formally defined in first-order logic, and axioms in FOL that enable us to deduce "predicates" based upon Predicate Logic [Cawsey 97]. Hence, to do the task we need to ensure we have the predicates required to do the task. The terms, predicates and axioms to do the task are summarized in Table 5.

TABLE 5 Micro-Theory of Resource Cost Units: Terms, Predicates, Axioms

Terms	*Predicates*	*Axioms*
Significant Resource	significant_res (r)	(FOL 84)
External Resource	external_res (r)	(FOL 50)
Internal Resource	internal_res(r)	(FOL 49)
Significant Activity	significant_act (a)	(FOL 85), (FOL 86)
Frontier Activity	frontier_act (a)	(FOL 61)
Internal Activity	internal_act(a)	(FOL 62)
Period Cost External Resource	period_cost_ext_res(r)	(FOL 97)
Non-Period Cost External Resource	non_period_cost_ext_res (r)	(FOL 103)
Non-Period Cost Internal Resource	non_period_cost_int_res (r)	(FOL 107)
Period Cost Internal Resource	period_cost_int_res (r)	(FOL 109)
Unit Cost of External Resource	unit_cost_ext (c, r)	(FOL 56)
Unit Cost Internal Resource	unit_cost_int (c, r)	(FOL 55)
(activity class): Specification Activity	activity(spec_act)	(FOL 76)
(relational term) - Specifies	specifies(spec_act, a)	(FOL 78)

In Sec.5.5, the cost behaviour for an activity instance was discussed to give insight into the committed resource cost unit, the enabled resource cost unit, the disenabled resource cost unit and the reenabled resource cost unit. We concluded that the committed resource *cost unit leads us to deducing the enabled resource cost unit and the disenabled resource cost unit of a resource. The committed resource cost unit should be computed based upon the concept of a commit_envelope of activities due to Resource Probing. In like manner, the reenable resource cost unit should be computed based upon the reenable_envelop of activities due to Resource Probing of a disenabled resource. Hence, in order to deduce resource cost units of a resource, it is sufficient that we establish the deduction of the committed resource cost unit of a resource.*

Given that one has a populated and instantiated frontiered activity model in TOVE, one is able to deduce the committed resource cost units for external and internal resources of the enterprise. The logical framework to do the required deductions is shown in Figure 19. presents the logical framework to deduce committed resource cost units for external and internal resources.

Essentially, Figure 19 shows that all committed resource cost units can be "automatically deduced" from a populated instantiated and frontiered activity model of an enterprise. The Micro-theory of ABC is deployed within the framework as the cpr's, cpa's, and cpo's are being used. Also, since the model is populated with "Formalized Overheads", the costs of direct, indirect and overhead costs are used to compute the unit cost of each "significant" resource being cycled through the framework.

The "macro-steps" rationalized in Figure 19 towards the deduction of committed resource cost units are:-

1. Start with some rknown resource of activity. Test for significance. Insignificant resources are stored aside, and the next rknown processed.
2. Deduce if resource is an external resource or an internal resource.
3. If it is an external resource, deduce its unit cost, its committed resource cost unit, and its cpr for an internal activity. Process the activity to step 8 below to continue.
4. If it is an internal resource, deduce the activity producing it from the produce_spec of Resource Ontology (Fadel et al.94).

5. Deduce if the activity of step 4 is frontier activity or internal activity.

6. If the activity of step 4 is frontier activity, deduce its external resources from use_spec and consume_spec of Resource Ontology. Since unit costs are known for external resources, the cpr's of external resources are deduced. The cpa of the frontier activity is deduced as being the sum of cpr's of its external resources.

7. If activity of step 4 is an internal activity, deduce all its resources from use_spec and consume_spec as before. Process all its external resources, deduce the committed resource cost units of external resources, deduce cpr's of its external resources. Deduce internal activities or frontier activities where required. If an internal activity go to step 5. If an external activity go to step 6.

8. Once the committed resource cost unit of a resource is deduced, the internal or frontier activity where required is deduced and checked for significance. An insignificant activity is stored aside, and the next rknown is processed. A significant activity, whether it is frontier or internal, is then processed for each of its resources to get the committed resource cost unit.

9. All frontier activities are costed first through step 6. Next internal activities requiring external resources and internal resources are costed. Eventually all internal activities requiring internal resources only are costed. The costing works from the "frontier or boundary envelope through to the core" much like an onion being peeled layer by layer. Each layer is costed until the final core or cost object is reached.

The framework of Figure 19 can be deployed within the "Reenable Envelope" and the "envelope of activities that contribute to specifying an activity".

5.12.3 Deducing "Open Book" Costs with ABC

Figure 20 represents a framework of Axioms and Formalized Concepts that compose the Theory of Resource Cost Units. Essentially, it serves as a "check tool" to ensure that the research has composed "formalized axioms" that enable an enterprise to deduce costs with

ABC, given that we have on hand a populated model, referred to in this research as Enveloped Activity Based Enterprise Model (EABEM).

The framework considers direct, indirect and overhead costs which are included in the committed resource cost units. Figure 20 promotes "open book" [Ruhl] costing with ABC.

FIGURE 19 Deducing Committed Resource Cost Units in Populated Model

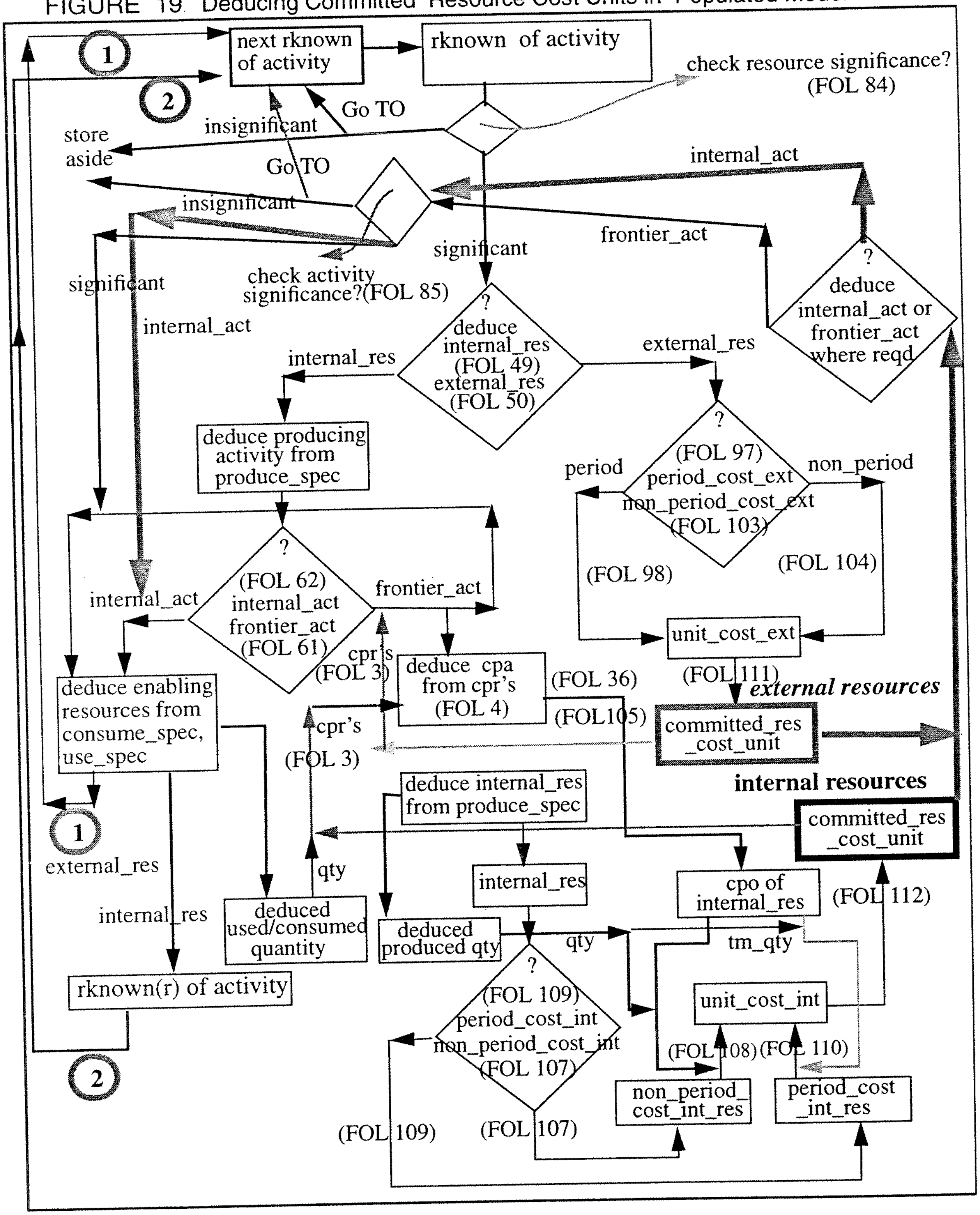
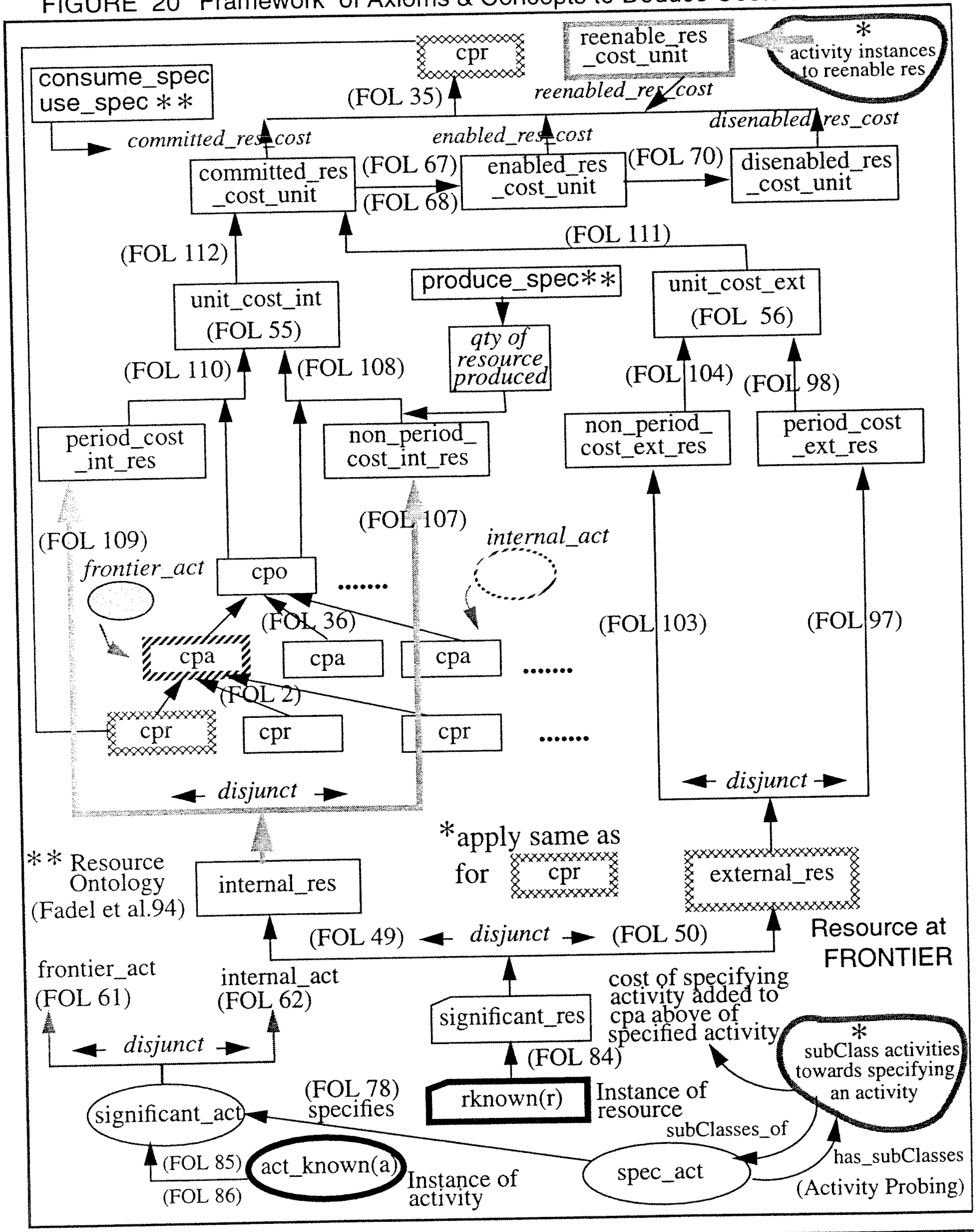


FIGURE 20 Framework of Axioms & Concepts to Deduce Costs with ABC



5.13 Concluding Remarks

The Cost Ontology for TOVE and Micro-theory for ABC in the previous chapter formalized the cost point of resource (cpr), the cost point of activity (cpa), and the cost point of cost order (cpo) so that the cost of a cost order (or cost object) of an activity may be deduced through ABC. This formalization of ABC was premised on knowing the resource cost units of each resource of an activity.

A Theory of Resource Cost Units in this chapter formalizes the procedures and axioms to deduce the resource cost units of resources. This theory is based upon:-

- Assumptions
- Taxonomy and Definition of Terminologies
- Principle of Resource Probe
- Commit Envelope of an Internal Resource
- Reenable Envelope of a Resource
- Formalization of Overheads
- Defining a Enveloped Activity Based Enterprise Model (EABEM)
- Micro-theory to deduce Resource Cost Units.

From an enterprise modelling perspective, the enterprise modeller must establish an Enveloped Activity Based Enterprise Model (EABEM). In order to guide the model builder towards EABEM, the Principle of Resource Probe has been formalized so that the probes of resources terminate at frontier activity clusters. Consistent with CAM-I, EABEM is composed of significant activities and significant resources as asserted by the enterprise.

The Activity, Resource and Cost Fluent Taxonomies presented in this chapter form a framework for a Formalization of Overheads. This framework enables us to visualize, and to embed a representation of knowledge within EABEM to "traditional overhead entities" such as depreciation, property taxes, cost of borrowing, insurance, heat, light, etc. so that the high cost of overheads in today's enterprises are included in the cost of each activity instance. Though no specific reference is made to high "information technology (IT) costs" of today, the developments of this chapter could accommodate this class of "IT overhead cost" through items of Table 2 such as *equipment depreciation*, *borrowed capital cost*, *leases*, *management supplemental benefits*, and *salaries* of "knowledge workers". This ensures that the appropriate, accurate and consistent inclusions of overheads to cost objects of activities are achieved in accordance to the ABC paradigm. Overheads are directly traceable to activities at "point of requirement". Overheads are allocated to activities based upon actual activity time utilization or actual quantity utilization as required. A framework of "open book" costing has been presented, where the complexities of "direct, indirect, overheads, and all 'sorts' of drivers" have been removed.

The framework of Figure 19 makes it possible to deduce the unit costs of resources, the essential ingredient to ABC. Figure 19 provides the answer to the fundamental question for "ABC-ers": From where and how does one get the unit cost of a resource?

The following two chapters use some of the Cost Ontology of Ch. 4 and some of the developments of this chapter to implement a prototype for a Cost Advisor so as to answer informal competency questions for case studies at the deHavilland and BHP enterprises.

CHAPTER 6 Forensic Cost Analysis with ABC at deHavilland

6.1 Introduction

This document describes a forensic cost analysis with ABC that spans the conventional notions of "direct, and overhead (or indirect) activities" for a model created for deHavilland Inc.(Bombardier), Toronto.

From a practical perspective, highlights of this study with reference to representing and reasoning about costs with enterprise models and ABC are:-

1. It promotes the trace of "overheads" directly to "point of consumption", i.e., an activity, and the reason for doing so. This eliminates the use of "drivers". ABC developments to date use "drivers" that are supposed to "indirectly trace" overheads to activities, and reasons for their usage are usually unclear. (Figure 23 clearly displays the activity cluster which "consumes" depreciation, insurance, property taxes, and utilities; and the "reason or cost object" for doing so).
2. Allocation or distribution of "fixed cost overheads" such as depreciation, property taxes, insurance, etc. are made to products based upon actual plant utilization. (Table 7 and the ensuing comments on page 186 with regard to the Formalization of

Overheads of Ch. 5 serve to illustrate this allocation for depreciation costs.) This allocation promotes change management towards "idle capacity". Higher the "idle capacity", higher will be the overhead cost component in the product price. According to Shields & McEwen [96]: "An important reason for unsuccessful implementations of ABC is that many companies have emphasized the architectural and software design of ABC systems at the expense of behavioral and organizational issues."

3. The final cost of the cost object is inclusive of direct, indirect and overhead costs. This cost can be obtained as an answer to a simple query. Further, this final cost is broken down into sub - costs due to "idleness or dormancy", "true utilization or execution", and most importantly, "down-time" and "re-start ups".(This is illustrated in Figure 29). Also, along the way to the final cost, any resource may be reviewed at any time in text form (illustrated in Figure 25) and graph form (illustrated in Figure 27) for "idleness" (or committed) cost, true utilization (or enabled) cost, resource down-time (or disenabled) cost, and for the costs to re-commission (or reenable) a resource. This is invaluable to companies in seeking opportunities for capital investments.

6.2 Brief Background about the Case Study

The study was undertaken over a period of approximately 6 months in 1996 and involved regular weekly visits that called for an average of 10 hours per week on site. *The "competency questions" surrounding the case study are real.* The Cost Ontology and Micro-theory of ABC deployed for a Cost Advisor to provide solutions was based upon Ch. 4. However, the path to the answers and the answers to the questions have been shortened and altered to a great extent; first, due to the confidential and sensitive financial and operational data originally used; and second, due to some technical limitations encountered with OAK in the TOVE Testbed. Data was assembled based upon observations made during site visits, interviews, questionnaires, group meetings, and the analysis of 1995 operations' and finance data. For the most part, activity and resource nomenclature are true to deHavilland. The original model had 56 classes of activities and approximately 700 plus activity instances. Any reference to "actual data" within this chapter from here on, should *not* be construed as meaning "the real data".

The formalization of "overheads" in Ch. 5 has made it clear that there is no distinction between overhead (or indirect) and direct activities. Hence, the Cost Ontology and Micro-theory for ABC in TOVE makes no distinction between overhead costs and direct costs. In formulating overhead entities through a resource taxonomy from a cost perspective, Chapter 5 presented a strong case that much of the traditional notions of overhead are directed from resource to activities, and via the activities, direct to cost orders (products, services, customers).

The objective of doing a forensic cost analysis at deHavilland is to deduce a more accurate cost of one of their products - *cured leading edges* - given that events have already occurred and some actual cost and event data are available to us. The given data will enable us to compute some initial resource cost units of frontier resources. The Cost Ontology and Micro-theory of ABC are then applied to deduce the costs of activities and cost of the cost order or cost object, viz., cured leading edges.

6.3 Background: deHavilland Inc. (Bombardier)

DeHavilland is one of the aircraft manufacturing divisions of the Bombardier group of companies. The company produces the Dash Series (100, 200, and 300 series) of twin turbo-prop engine aircrafts widely used all over the world. Recently, the company has also begun producing the 400 series to accommodate more passengers than the 300 series, and the Global Express that uses the Rolls Royce jet engines.

The company occupies over one million square feet of manufacturing facilities and administrative offices in the heart of metropolitan Toronto (Downsview) of the Province of Ontario, which is the most industrialized province in Canada. The company has over fifty years of experience in aircraft production. It is a unionized shop and employs approximately 5,800 personnel. With this work-force complement and facilities, it is estimated that maximum plant capacity is about 5 aircraft per month.

6.3.1 Motivating Scenario

It has been identified that the company is experiencing problems in applying strategic product costing to a specific class of products called Leading Edges for their aircraft wings. This product mainly falls within the Methods Fabrication Department of deHavilland.

As the name Leading Edge implies, this product comprises the front edge or leading edge of an aircraft wing. The Production Process Standards (referred to as PPS at deHavilland) for leading edges calls for highly specialized raw materials such as reinforced fibre-glass composite sheets that have to be intricately cut and laid up for specialized curing and forming processes that demand highly skilled and costly labour. After several impregnations of epoxies and resins followed by curing processes that use highly specialized and capital intensive *autoclave equipment*, the initial cloth like composite fabrics end up as a very hard laminated, rounded and elongated shaped product called the *cured leading edge*. This product then goes through a paint line procedure and eventually forms the front portion of an aircraft wing. The finished leading edge includes the "de-icer boot" system made up of specialized rubber. Through this de-icer boot system, tiny jets of warm air are pumped in order to "de-ice" the wings. The integrity of this system is absolutely essential for ensuring that the aircraft wings are de-iced adequately to ensure safe take-off and landing of the aircraft.

Their product of concern is the cured leading edges. In the context of deHavilland, strategic product costing is considered a process of relating costs to cured leading edges in ways that help the company form and review strategy for the manufacture of the product. The company has recognized that ABC is one of the primary ways in which strategic costing may be implemented at deHavilland.

DeHavilland is interested in implementing ABC for cured leading edges for the following reasons:-

1. The company wishes to have a costing process in place to support strategic decision making pertaining to leading edges. The strategies concern:-

- understanding and computing true overheads or indirect costs to cured leading edges rather than the arbitrary 600 - 700 percent allocation of overhead costs to leading edges based on the large proportions of direct labour or machine hours used,
- formulating integration strategy for "make versus buy" by understanding and computing true product line profitability for leading edges.

2. Owing to high sales growth in the Asian market, the company has made some long-term commitments towards the selling price of their aircrafts. In the past, the company tended to consider those costs with small changes or costs that will not change much in the short-run as being fixed and not relevant to product profitability. However, deHavilland realizes that for decisions involving a long-term commitment to a selling price, those so-called fixed costs do not stay fixed. Consequently, the company is faced with the challenge to discover and measure the links between strategic decisions that commit to a selling price to a customer and those "fixed" costs because such costs tend to be indirect, in the traditional sense of being not clearly traceable to products. deHavilland finds itself faced with the task to discover longer term variability and some degree of cause-and-effect connections throughout all their products.

3. The scope of relevant costs to deHavilland is not limited to manufacturing costs. Being a large company, the company incurs high costs due to personnel administration, personnel supplementary benefits, cost of capital, costly equipment depreciation, insurance and the high Toronto land taxes. These are some of the relevant costs that the company wishes to trace to their product lines. They do NOT wish to consider cost relevance by what costs can be inventoried as per following General Accepted Accounting Practices (GAAP), but by the identification of links between strategies adopted for leading edges and costs traced to leading edges.

Based on some of the motivations mentioned, the company desired to have a forensic cost analysis done with ABC on the cured leading edge product. Their objective is to identify all "direct and indirect" cost entities traceable to the product, and to deduce more accurate cost of the product given that events have already occurred and some actual cost and event data are available.

6.4 Competency Queries to be Solved

deHavilland was interested in the following competency questions based upon 1995 historical data that required some forensic analysis. These questions are stated in natural language as follows:-

1. What should the company consider as being the 1995 ABC cost per cured leading edge?
2. Based upon an average autoclave activity instance, what should the $ amount be that the company allocates towards factory space depreciation associated with the activity?
3. How many dollars of property tax expenditures should be allocated towards the autoclave activity instance producing cured leading edges?
4. What should be the overhead costs associated with the 2,650 square feet of space utilized by the autoclave activity producing leading edges?

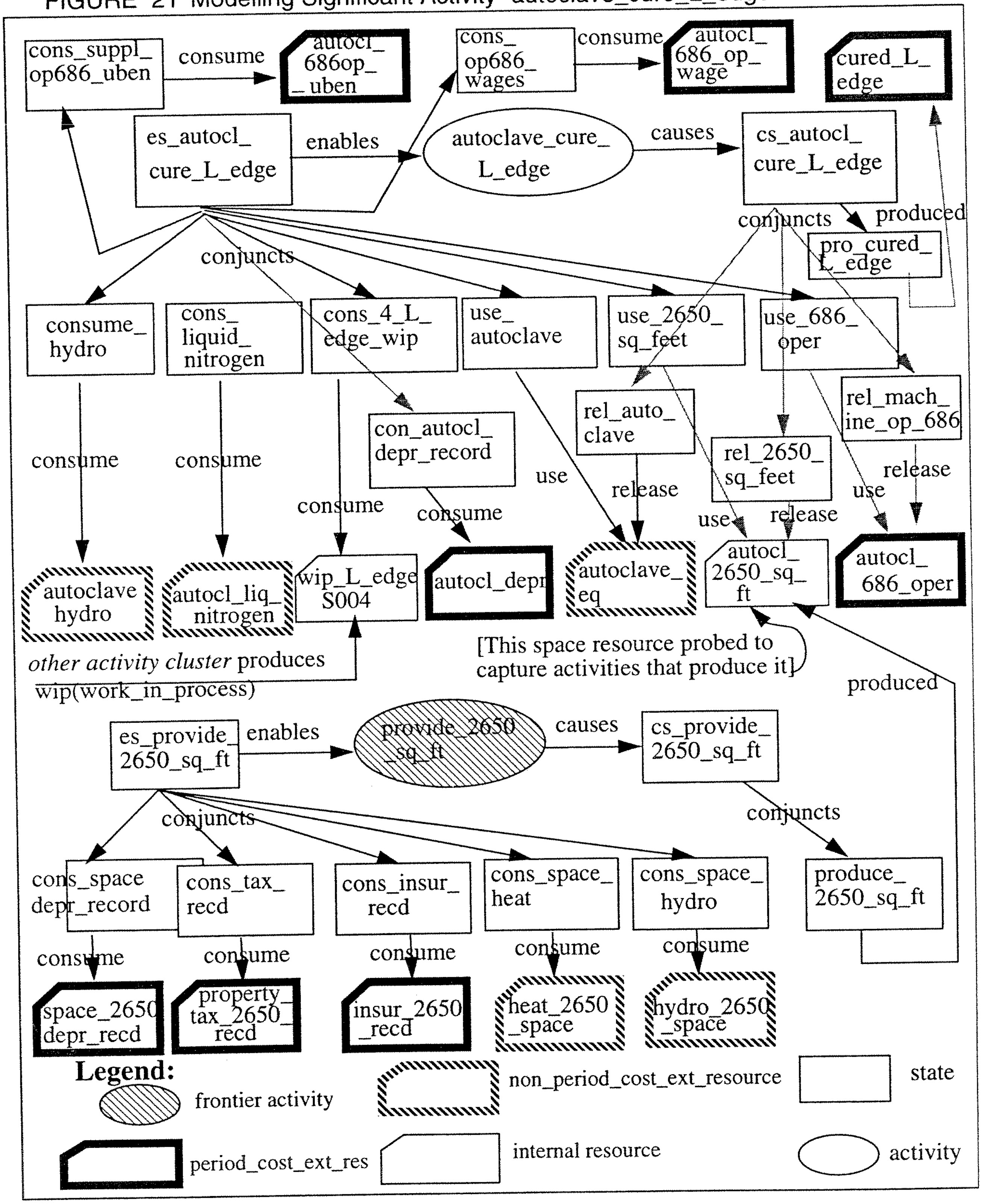
In order to solve these queries through a forensic cost analysis of 1995 data, an enveloped activity based model, together with a ROCK and Prolog data base had to be built in the TOVE Testbed environment that uses the graphical user interface software tool called OAK **[EIL 94]**. The TOVE Testbed environment, the OAK user interface, the FOL axioms implemented in Prolog, together with the ROCK and Prolog knowledge data base form the frame work of the Cost Advisor through which solutions to queries are sought.

6.5 Preliminaries of Enterprise Modelling Prior to Postmortem Cost Analysis

Any reference to "activities" implies the activity cluster in TOVE. A populated and instantiated activity model refers to the various classes of activities encountered at deHavilland, together with several instances of the various class activities. The deductions of costs with the Cost Ontology and Micro-theory of ABC are made at the instance level.

Initially, the classes of activities and instances of activities were loaded as data was being assembled. Based upon the developments of Ch. 5 regarding significant activity, significant resource, frontier activity, internal activity, Resource Probing, and "formalized overheads", an Enveloped Activity Based Enterprise Model (EABEM) was identified for deHavilland through the deployment of axioms of Figure 20 in Ch. 5. As stated earlier, the original EABEM was fairly extensive, i.e., the breadth of class activities included 56 classes and spanned over 700 instances. In other words, there were many layers of internal activities until the frontier activities were reached. The model presentation in this chapter has been greatly "shrunk" - it extends over 8 classes of activities. In other words, the frontier is brought artificially closer to the final cost object, the cured_leading_edge. Also deHavilland was not interested in "cost of specifying activities" as it fell outside the jurisdiction of their "enterprise". Records indicated that there was no "commit interval" for resources. In other words, the committed resource cost unit of a resource is taken to be equivalent to its enabled resource cost unit.

FIGURE 21 Modelling Significant Activity "autoclave_cure_L_edge"



Refer to activity cluster of figure 21. The frontier or external resoures required by an instance of significant activity autoclave_cure_L_edge as shown in Table 1 are:- hydro (autoclave_hydro), liquid nitrogen (autocl_liq_nitrogen), autoclave equipment depreciation (autocl_depr), autoclave equipment (autoclave_eq), a unionized Class 686 operator (autocl_686_op), the supplemental union benefits for the 686 operator (autocl_686op_uben), and the 686 operator wages (autocl_686op_wage). The two internal resources required by the autoclave_cure_L_edge activity instance are the 2,650 square feet of factory space (autocl_2650_sq_ft) and the work-in-processed type S004 leading edges (wip_L_edge_S004) that are loaded into the autoclave equipment to be cured.

6.6 Using Historical Data & Deducing Costs

The activity model development and the extent towards achieving a frontiered or enveloped activity based enterprise model for cured leading edges have been guided with the Principle of Resource Probing and the formalization of traditional notions of overheads within the Theory of Resource Cost Units. The forensic cost analysis is made possible by obtaining deductive solutions to queries about cost points of resources (cpr), cost points of activities (cpa) and cost points of orders (cpo) based on the Cost Ontology for Enterprise Modelling and the Micro-theory for costing based on the ABC paradigm.

Guided by the Principle of Resource Probing and the Taxonomy of Resources in the Formalization of Overheads, the activity "provide_2650_sq_ft" is identified that notionally produces 1 unit of the resource, autocl_2650_sq_ft, that is required for each instance of the autoclave_cure_L_edge activity. This activity instance is a frontier activity and notionally requires frontier resources identified in Table 6.

TABLE 6 Data for Significant Activity "autoclave_cure_L_edge"

Activity Name	*Total Actual Activity Time for 1995*	*External (E) & Internal (I) Resources Required*	*1995 Estimated Delay or Ineffective Time*	*Avg. load per instance*	*Average Actual Instance completion time (Hrs)*
autoclave_cure_ _L_edge	1, 780 hours*			1 L_edge (S004 type)	6 Hours **
		autocl_2650_sq_ft (I)			
		wip_L_edge_S004 (I)			
		autoclave_hydro (E)			
[Internal Activity]		autocl_liq_nitrogen (E)			
		autocl_depr (E)			
		autoclave_eq (E)			
		autocl_686_oper (E)			
		autocl_686op_uben (E)			
		autocl_686op_wage (E)			

* The Total Actual Activity Time for 1995 is 1,780 hours.

** The 6 hours completion time per activity instance is an average based upon 10 instances.

TABLE 7 Resource Cost Units for Frontier Activity "provide_2650_sq_ft"

Activity Name	*Total Actual Activity Time for 1995*	*External Resource*	*Resource Class*	*Actual $ Costs for 1995*	*Enabled Resource Cost Unit*
provide_2650_sq_ft	1,780 hours based on auto-clave_cure__L_edge	space_2650_depr_rcd	period_cost_ext_res	$5,350 (space depreciation cost)	$5350 / 1780 ~ $3 / hr *
		property_tax_2650_recd	period_cost_ext_res	$ 3,575 (factory space property tax cost)	$3575 / 1780 ~ $2 / hr *
		insur_2650_recd	period_cost_ext_res	$ 3,400 (factory space insurance cost)	$3400 / 1780 ~ $2 / hr *
		heat_2650_space	non_period_cost_ext_res	$980 for 1000 cubic meters	~ $1/cu.m
		hydro_2650_space	non_period_cost_ext_resource	$ 950 for 1000 kwh	~ $1/kwh

[* The factory space of 2650 square feet is dedicated only to instances of activity autoclave_cure_L_edge which had a total activity duration time of 1,780 hours for the 1995 year. The activity provide_2650_sq_ft notionally sustains the autoclave_cure_L_edge activity by satisfying the space requirement of the latter as needed. Hence, fixed costs of depreciation, property tax and insurance are considered absorbed by or distributed over the 1,780 hours of the autoclave_cure_L_edge activity for the 1995 year.

For example depreciation is represented by space_2650_depr_rcd, according to the Formalization of Overheads in Ch. 5, Table 2 calls for Nonactivity_cost Fluent bldg_depr_cost(tps, tc_ext, tt_act, r) where, in this instance, tps = 1year, tc_ext = $5, 350 per year, tt_act = 1,780 hours, r = space_2650_depr_rcd. Then applying, Axiom (FOL 98), the unit cost for depreciation = ($5, 350 / 1 year) (1 year / 1, 780 hours) = $3 / hour. Notice, if plant utilization is decreased, the unit cost of depreciation would increase.

Therefore, the cost component of building depreciation toward product cost would increase with decrease in plant utilization. In other words, product cost is sensitive to and bears an inverse relationship to idle plant capacity.]

6.6.1 Deducing Costs of Resources and Activities

The activity, provide_2650_sq_ft, is identified as producing the internal resource, autocl_2650_sq_ft, with the guidance of the Principle of Resource Probing. The resource requirements for the activity must be satisfied to make possible the supply and usage of the factory space for the activity, autoclave_cure_L_edge. Based on the Cost Ontology and micro-theory for ABC, Table 8 summarizes the deduced cost of the resources required for the provision of factory space towards the production of cured leading edges.

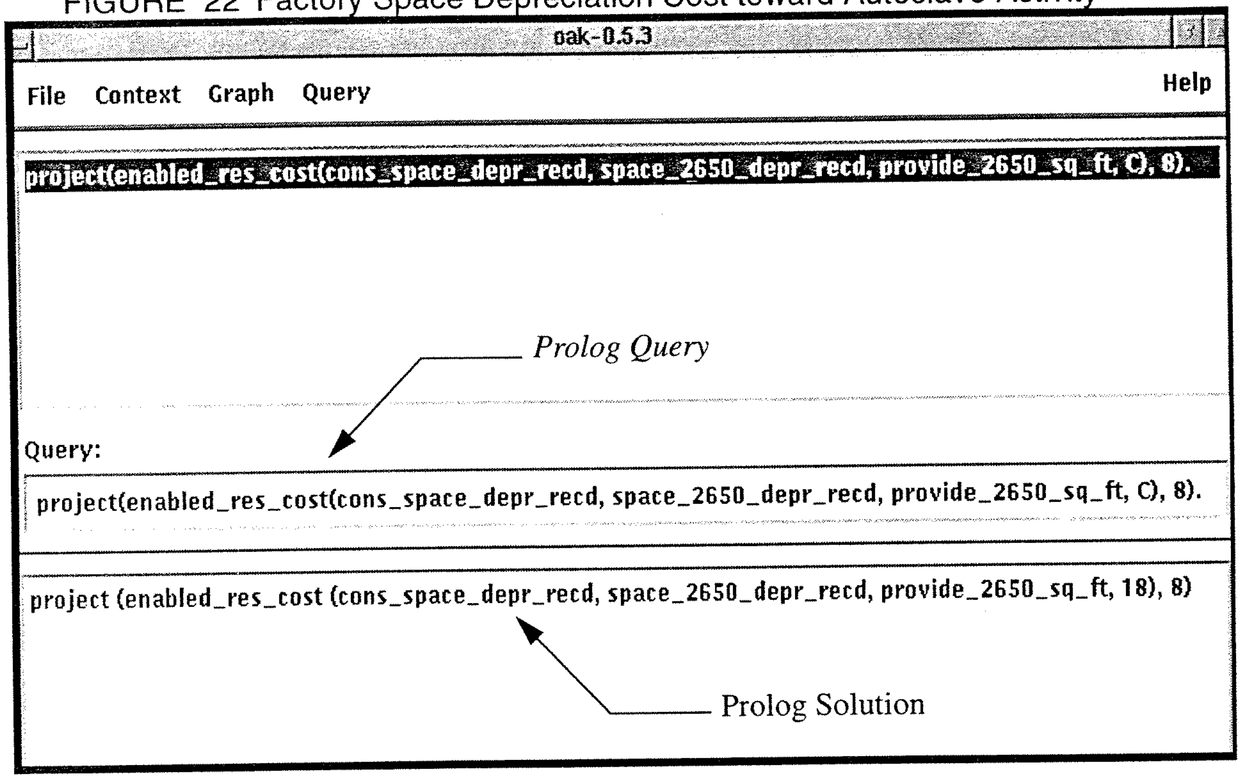
TABLE 8 Cost of Resources Required for each Instance of activity "provide_2650_sq_ft"

Activity Name	*External Resource*	*Resource Class*	*Resource consume specs. (qty required)*	*Enabled Resoure Cost Unit*	*Enabled Resoure Cost of resource required*
provide_2650_sq_ft	space_2650_depr_recd	period_cost_resource	1 hr/hr	~ $3 /hr	$ 18 *
	property_ tax_2650_recd	period_cost_resource	1 hr/hr	~ $2 /hr	$ 12
[Frontier Activity]	insur_2650_recd	period_cost_resource	1 hr/hr	~ $2 /hr	$ 12
	heat_2650_space	non_period_cost_resource	3 cu.m / hr	~ $1/cu.m	$ 18
	hydro_2650_space	non_period_cost_resource	1 kwh / hr	~ $1/kwh	$ 6
				TOTAL COST	$ 66 / unit
	Resource produced ....autocl_2650_sq_ft				

* Note: The average time duration of the autoclave_cure_L_edge activity instance was 6 hours. Hence, ($3/hr x 6 hrs) = $18

Figure 22 illustrates the Prolog query input and solution obtained re: the factory space depreciation cost towards the autoclave activity. The interpretation of the Prolog solution indicates that the factory space depreciation cost towards each instance of the autoclave activity for the 1995 year is $18 based upon the actual historical data for the 1995 year.

FIGURE 22 Factory Space Depreciation Cost toward Autoclave Activity

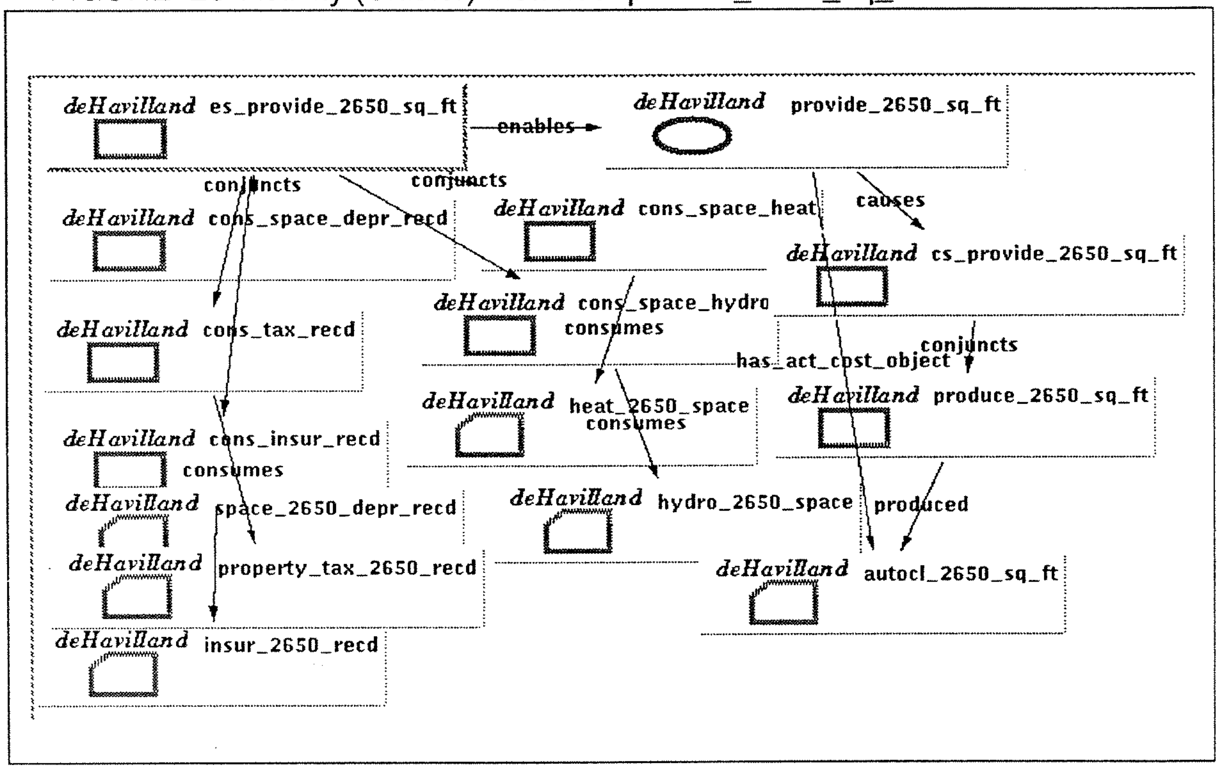


In like manner,

- *property tax cost* per activity instance = ($2/hr) x (1 hr / hr) x (6 hrs) = $12
- *insurance cost* per activity instance = ($2/hr) x (1 hr / hr) x (6 hrs) = $12
- *heating space cost* per activity instance = ($1/cu.m) x (3cu.m/hr) x (6 hrs) = $18
- *hydro (lighting) space cost* per activity instance = ($1/kwh) x (1 kwh/hr) x (6 hrs) = $ 6

The activity based cost of producing 1 unit resource "autocl_2650_sq_ft" for instance of activity "provide_2650_sq_ft" = the cost of all resources required for the completion of the activity instance = $18 + $12 + $12 + $18 + $6 = $66 per 1 unit of resource "autocl_2650_sq_ft".

FIGURE 23 Activity (Cluster) Instance "provide_2650_sq_ft"



The visualization of the deHavilland activity cluster "provide_2650_sq_ft" on the TOVE Testbed is shown in the Figure 23 [interpreted as follows:- activity provide_2650_sq_ft is contingent upon (enabling states) property taxes, insurance, space depreciation, heat, light, etc. being "paid for" (consumed) in order to "cause" the production of the resource (cost object) autocl_2650_sq_ft in the caused state of the activity]. A portion of the ROCK knowledge database that consists of class_frames with attributes and relations applicable to Figure 23 and Table 8 is shown in Figure 24.

FIGURE 24 Rock data base

```
class_frame provide_2650_sq_ft
{subclassOf:dehav_significant_act;
attributes:
actual_total_activity_time = 1780 ;
activity_instance_time = 6 ;
relations:
enabled_by = es_provide_2650_sq_ft ;
causes = cs_provide_2650_sq_ft ;
has_act_cost_object = autocl_2650_sq_ft;
}

class_frame es_provide_2650_sq_ft
{subclassOf:state_discrete_conjunction ;
relations:
conjuncts = {cons_space_depr_recd, cons_tax_recd, cons_insur_recd,
cons_space_heat, cons_space_hydro };
}

class_frame cs_provide_2650_sq_ft
{subclassOf:state_discrete_conjunction ;
relations:
conjuncts = produce_2650_sq_ft;
}

class_frame cons_space_depr_recd
{subclassOf:discrete_consumption;
relations:
consumes = space_2650_depr_recd;
attributes:
cons_spec_rate = 1;
}
class_frame space_2650_depr_recd
{subclassOf:period_cost_ext_res;
attributes:
enabled_res_cost_unit = 3;
actual_total_cost = 5350;
total_quantity = 5;
}
```

The state of the resource "autocl_2650_sq_ft" is typically always enabled for activity instance "autoclave_cure_L_edge" to execute. Therefore, in reasoning the costs for activity instance "autoclave_cure_L_edge",

- the commit resource cost unit of resource "autocl_2650_sq_ft" = the enable resource cost unit of resource "autocl_2650_sq_ft" = $66 per 1 unit of resource "autocl_2650_sq_ft".

A portion of the Prolog knowledge base for the input of the instance of activity provide_2650_sq_ft is as follows assuming the activity occurs at time point 1 and the activity is completed after 6 hours at time point 7:-

```
:- multifile holds/2, occursT/2, concurrent/2.
% activity_occursT (provide_2650_sq_ft, 1).
occursT (enable (cons_space_depr_recd, provide_2650_sq_ft), 1).
occursT (enable (cons_tax_recd, provide_2650_sq_ft), 1).
occursT (enable (cons_insur_recd, provide_2650_sq_ft), 1).
occursT (enable (cons_space_heat, provide_2650_sq_ft), 1).
occursT (enable (cons_space_hydro, provide_2650_sq_ft), 1).
occursT (complete (cons_space_depr_recd, provide_2650_sq_ft), 7).
occursT (complete (cons_tax_recd, provide_2650_sq_ft), 7).
occursT (complete (cons_insur_recd, provide_2650_sq_ft), 7).
occursT (complete (cons_space_heat, provide_2650_sq_ft), 7).
occursT (complete (cons_space_hydro, provide_2650_sq_ft), 7).
...............
..............
concurrent (a, b).
```

[Note: In the above, since no time data is given about the commit, disenable or re-enabled time intervals, the commit, disenabled and re-enabled resource costs of each resource required by the activity would be deduced as being zero. This is shown later in this chapter].

Question:- How much space_depr_recd resource available at time point 1 (start of activity) and at time point 8 (end of activity)?

Prolog Query:- project_times(rp(space_2650_depr_recd,Q),[1,8]).

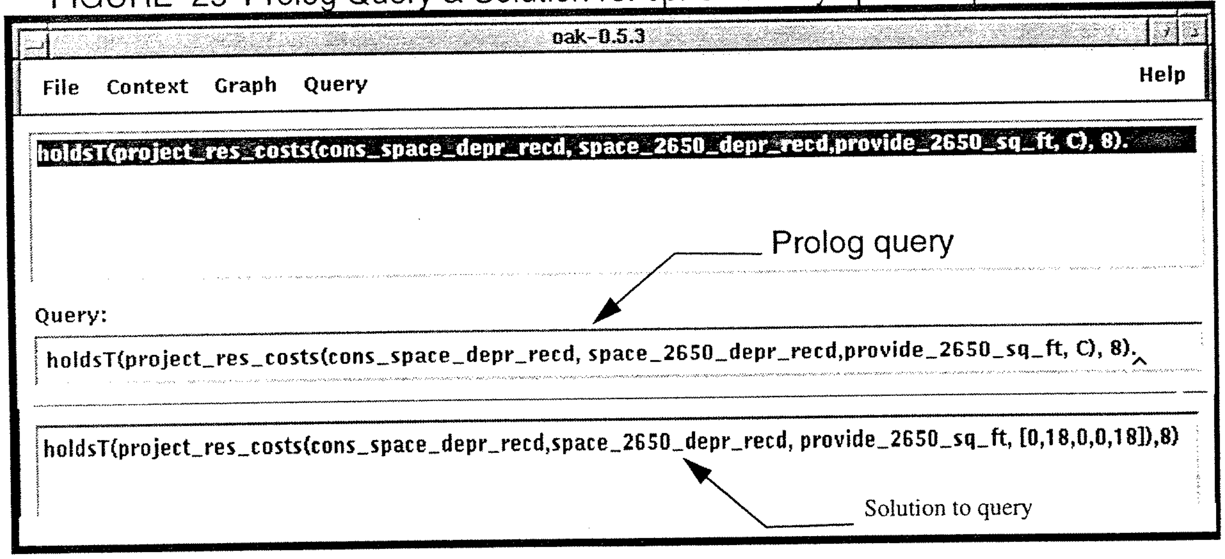
Solution: project_times(rp(space_2650_depr_recd,[5,4]),[1,8])

(The above solution indicates that at the start time point 1, 5 units of resource space_2650_depr_recd availabe; and at end time point 8, there are 4 units of resource as 1 unit depleted by the activity instance being completed. The activity instance consumed 1 unit out of the 5 units available. This solution is displayed graphically in Figure 26.

Question:- What is the cost of factory space depreciation per activity instance?

OR: How much is the cost of space_2650_depr_recd required by the end of the activity instance at time point 8? (Figure 25 indicates the solution to the query.)

FIGURE 25 Prolog Query & Solution re: cpr of factory space depreciation



For the resource, space_2650_depr_recd, the above solution .... [0, 18, 0, 0, 18], 8 .... indicates that the instance of the activity at time point 8 has commit resource cost = \$0, enabled resource cost = \$18, disenabled resource cost = \$0, re-enabled resource cost = \$0, and the total resource cost = \$18 equivalent to 0 + 18 + 0 + 0. The solution is displayed graphically in Figure 27.

FIGURE 26 Resource Points of “space_2650_depr_recd” at deHavilland

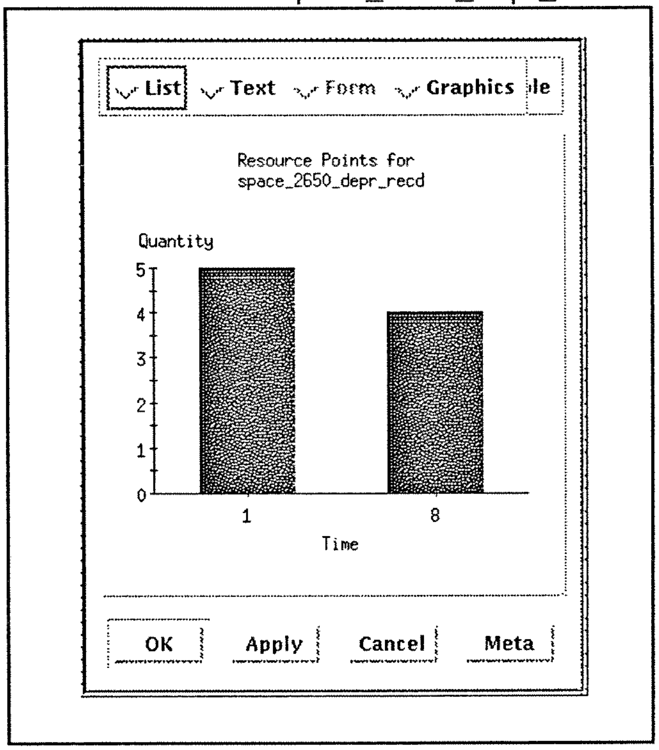
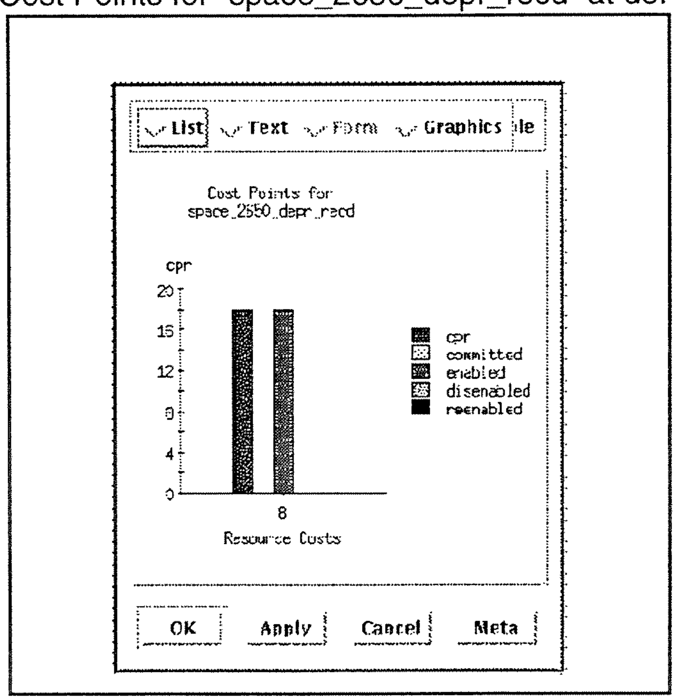


FIGURE 27 Cost Points for “space_2650_depr_recd” at deHavilland



6.6.2 Using Principle of Resource Probe

The probe of internal resource wip_L_edge_S004 (refer figure 21), leads to the identification of the following sub-actions of the commit envelope of the resource:-

1. activity sup_f_pts_L_edge_005 of time duration 2 hours, produces wip_L_edge_S004, and requires internal resource v_L_kit_w_psr to be probed (refer Table 13);

2. activity sup_v_pts_L_edge_004 of time duration 2 hours, produces v_L_kit_w_psr, and requires internal resource rm_kit_w_psr to be probed (refer Table 12);

3. activity sup_r_mtls_L_edge_003 of time duration 3 hours, produces rm_kit_w_psr, and requires internal resource psr_report to be probed (refer Table 11);

4. activity load_L_edge_p110_002 of time duration 2 hours, produces psr_report, and requires internal resource wkorder_drwg_L_edge to be probed (refer Table 10);

5. activity drwg_85720077_S004_cd_001 of time duration 4 hours, produces wkorder_drwg_L_edge, requires all external resources and is a frontier activity (refer Table 9).

Therefore, an accurate cost of the internal resource, wip_L_edge_S004, must reflect the costs of the above five activities that form the commit envelope of the resource, and which must be completed to make it possible to produce it. Each resource for the various activities in the deHavilland enterprise activity model towards the cost object, cured_L_edge, is queried to determine if a resource is an external or internal resource, and significant, in the deHavilland implementation. Summaries of resources that are external and internal are shown as part of Tables 9 - 13.

6.6.3 Costing a Frontier Activity

The activity, drwg_85720077_S004_cd_001, is a frontier activity by definition as all its resources are external resources as indicated in Table 9. This is the frontier activity for the internal resource, wip_L_edge_S004. From cost axiom schema 2 (Ch.4), the cost of an activity at a time point is equivalent to the sum of the cost points of the resources required by the activity at that time point. Given that the enabled states associated with each of the resources required by the activity, drwg_85720077_S004_cd_001, begin at time point 1 and is completed 4 time points (or 4 hours) later at time point 5. In the implementation, the cpa query about the cost of the completed activity at time point 6 (the closest time point beyond 5) is of the form:-

Prolog Query:- holdsT(project_act_costs (drwg_85720077_S004_cd_001, C), 6).

Solution: holdsT(project_act_costs (drwg_85720077_S004_cd_001, [0[1], 448[2], 0 [3], 0 [4], 448[5]]), 6)

[1] = dormant activity cost is $ 0

[2] = execute activity cost is $ 448

[3] = suspend activity cost is $ 0

[4] = reExecute activity cost is 0

[5] = activity cost = $ 448 = [$ 0 + $ 448 + $ 0 + $ 0]

The cpa solution of $ 448 for the activity drwg_85720077_S004_cd_001, confirms that it is the sum of the costs of the resources required by the activity as shown in table 9.

TABLE 9 Cost of Resources Required for each Instance of activity "drwg_85720077_S004_cd_001" ..... (Time = 4 hrs)

Activity Name	***Resource***	***Resource Class (E = external, I = internal)***	***Resource consume specs. (qty required)***	***Enabled Resource Cost Unit***	***Cost of resource required***
drwg_85720077_ S004_cd_001	eng_personA	E; non-period	1 unit	$ 0	$ 0
	salary _eng_personA	E, period	1 hr / hr	$30 / hr	$ 120
	eng_personB	E, non-period	1 unit	$ 0	$ 0
	salary_eng_personB	E, period	1 hr / hr	$35 / hr	$ 140
[Frontier Activity]	mgmt_benefitA	E, period	1 hr / hr	$ 10 / hr	$ 40
	mgmt_benefitB	E, periood	1 hr / hr	$ 12 / hr	$ 48
	wkorder_drwg_L_edge	E, non-period	1 unit	$ 100 / unit	$ 100
				TOTAL COST	$ 448 / unit
	Resource produced...wkorder_drwg_L_edge				

The frontier activity, drwg_85720077_S004_cd_001, produces 1 unit of resource, wkorder_drwg_L_edge, which is used by an internal activity. The commit and enabled resource cost unit of resource, wkorder_drwg_L_edge, is $ 448 per unit. From an ABC standpoint, the consumption of 1 unit of this resource by an activity would result in $448 being allocated to the cost of that activity.

6.6.4 Costing Internal Activities

Tables 9 through 13 summarize the deduced costs of each non-frontier activity that ultimately produces the internal resource, wip_L_edge_S004. The costs of each of these activities are deduced with Prolog queries as indicated previously.

TABLE 10 Cost of Resources Required for each Instance of activity "load_L_edge_p110_002" .....(Time = 2 hrs)

Activity Name	*Resource*	*Resource Class (E=External, I = internal)*	*Resource consume specs. (qty required)*	*Enabled Resource Cost Unit*	*Cost of Resource required*
load_L_edge_p110_002	op_936	E, non-perod	1 unit	$ 0	$ 0
	op936_wage	E, period	1 hr / hr	$ 25 / hr	$ 50
[Internal Activity]	op936_benefits	E, period	1 hr / hr	$ 8 / hr	$ 16
	wkorder_drwg_L_edge	I, non-period	1 unit	$ 448 / unit	$ 448
				TOTAL COST	$ 514
	Resource produced ....psr_report				

TABLE 11 Cost of Resources Required for each Instance of activity "sup_r_mtls_L_edge_003" .....(Time = 3 hrs)

Activity Name	*Resource*	*Resource Class (E = external, I = internal)*	*Resource consume specs. (qty required)*	*Enabled Resource Cost Unit*	*Cost of resource required*
sup_r_mtls_L_edge_003	psr_report	I, non-period	1 unit	$ 514 / unit	$ 514
	op595A_sup_r	E, non-period	1 unit	$ 0	$ 0
	wage_op595A_sup_r	E, period	1 hr / hr	$ 25 / hr	$ 75
	uben_op595A	E, period	1 hr / hr	$ 8 / hr	$ 24
[Internal Activity]	op595B_sup_r	E, non-period	1 unit	$ 0	$ 0
	wage_op595B_sup_r	E, period	1 hr / hr	$ 20 / hr	$ 60
	uben_op595B	E, period	1 hr / hr	$ 7 / hr	$ 21
				TOTAL COST	$ 694 / unit
	Resource produced ....rm_kit_w_psr				

TABLE 12Cost of Resources Required for each Instance of activity "sup_v_pts_L_edge_004" ..... (Time = 2 hrs)

Activity Name	*Resource*	*Resource Class (E = external, I = internal)*	*Resource consume specs. (qty required)*	*Enabled Resource Cost Unit*	*Cost of Resource required*
sup_v_pts_L_edge_ 004	rm_kit_w_psr	I, non-period	1 unit	$ 694 / unit	$ 694
	progress_opZ	E, non-period	1 unit	$ 0	$ 0
[Internal Activity]	wage_progress_opZ	E, period	1 hr / hr	$ 25 / hr	$ 50
	uben_progress_opZ	E, period	1 hr / hr	$ 8 / hr	$ 24
				TOTAL COST	$ 788
	Resource produced .... v_L_kit_w_psr				

TABLE 13Cost of Resources Required for each Instance of activity "sup_f_pts_L_edge_005" .....(Time =2 hrs)

Activity Name	*Resource*	*Resource Class (E = external, I = internal)*	*Resource consume specs. (qty required)*	*Enabled Resource Cost Unit*	*Cost of resource required*
sup_f_pts_L_edge_ 005	v_L_kit_w_psr	I, non-period	1 unit	$ 788 / unit	$ 788
	opY_f_pts	E, non-period	1 unit	$ 0	$ 0
	wages_opY_f_pts	E, period	1 hr / hr	$ 25 / hr	$ 50
[Internal Activity]	uben_opY	E, period	1 hr / hr	$ 8 / hr	$ 24
	mgrA_f_pts	E, non-period	1 unit	$ 0	$ 0
	mgrA_salary	E, period	0.25 hr / hr	$ 50 / hr	$ 25
	mben _mgrA	E, period	0.25 hr / hr	$ 16 / hr	$ 8
				TOTAL COST	$ 895 / unit
	Resource produced .... wip_L_edge_S004				

The enabled resource cost unit of the internal resource, wip_L_edge_S004, is $895/unit.

6.6.5 Deducing the Cost of the Cost Order for an Activity

The deHavilland enterprise activity model was initiated with the significant activity, autoclave_cured_L_edge. This activity produced the end product, cured_L_edge. Through this activity, and guided by the Principle of Resource Probing, the deHavilland activity based enterprise model was bounded by external resources.

From the ABC perspective, the deHavilland enterprise activity based model should confirm that the reason for the activity autoclave_cured_L_edge to be performed is the end product cured_L_edge. In referemce to the Cost Ontology, cured_L_edge is the cost order of the activity autoclave_cure_L_edge.

The activity model may be queried to find the cost order of the activity autoclave_cure_L_edge. This enquiry takes the following prolog query form and displays the output for the deHavilland implementation as shown in Figure 28:-

Prolog Query:- costorder_cluster(autoclave_cure_L_edge).

Solution: [visually displays the cost order of the activity as shown in Figure 28]

FIGURE 28 Querying about the cost order of activity

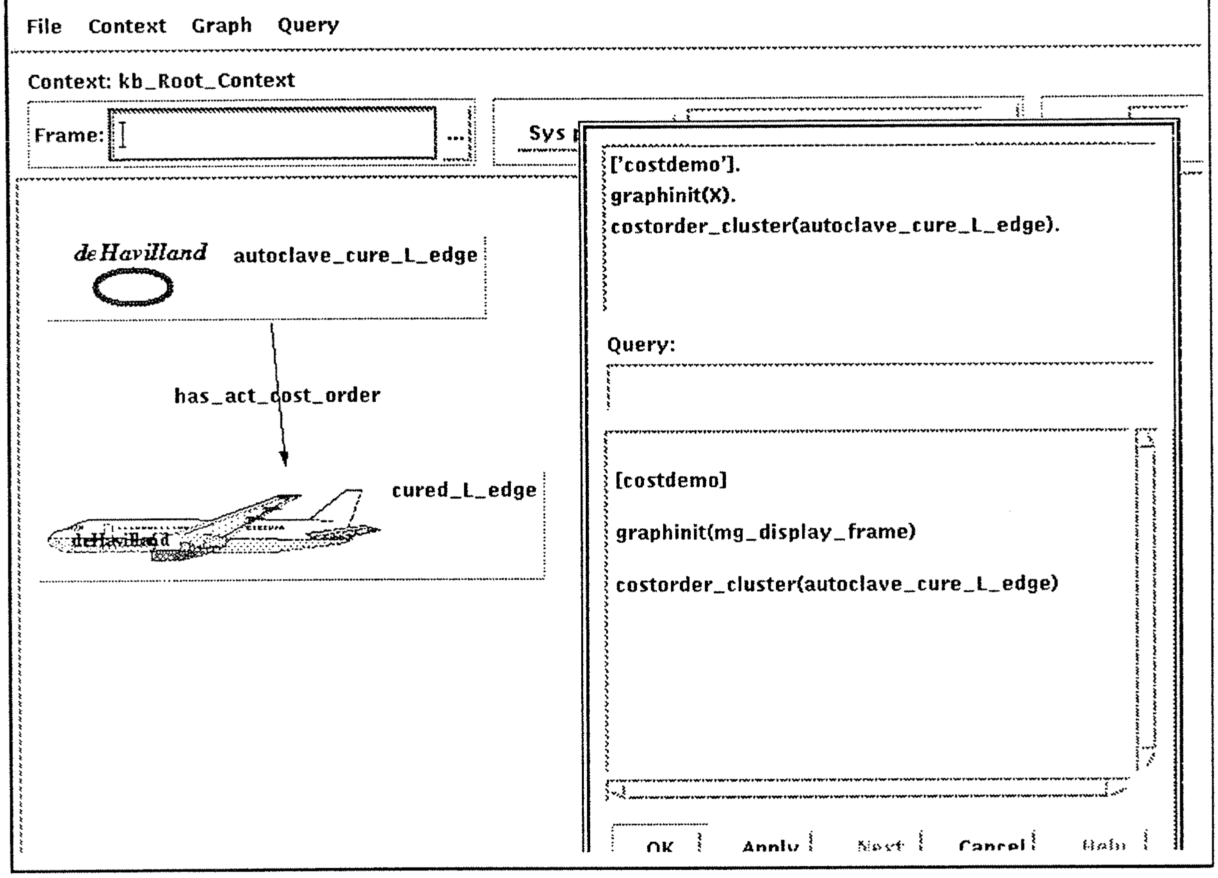


Table 14 shows a summary of the cost of resources: that is, cost point of resources or the cpr's of the resources, required by the activity instance, autoclave_cure_L_edge.

TABLE 14Cost of Resources Required for each Instance of activity "autoclave_cure_L_edge" ..... (Time = 6 hrs)

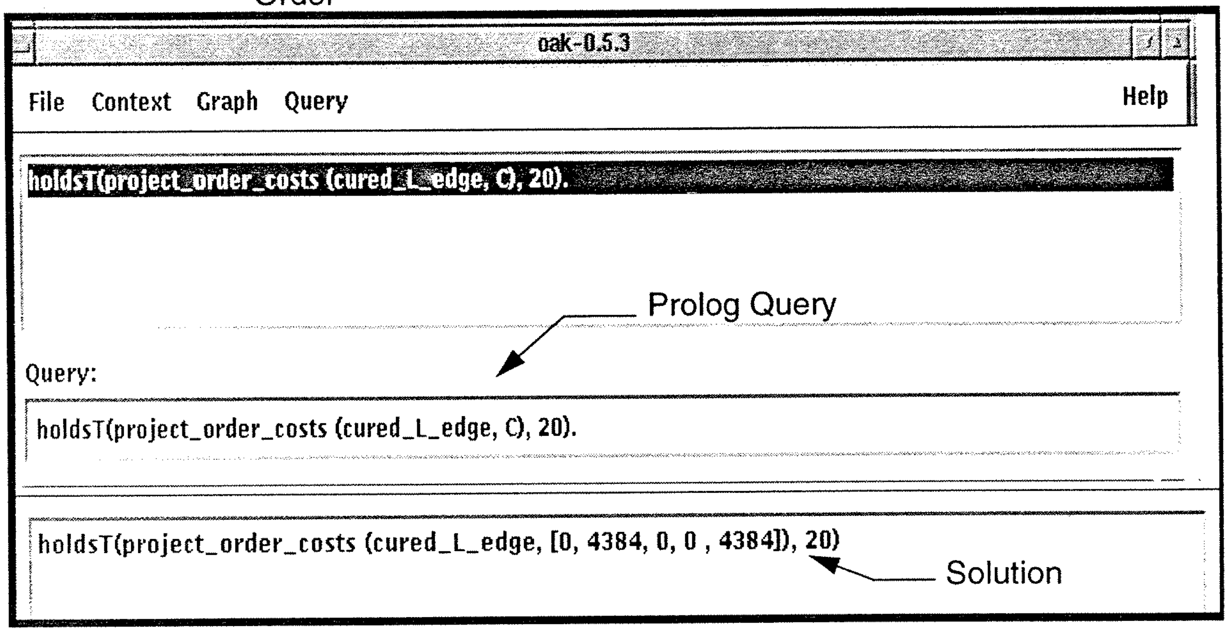
Activity Name	*Resource*	*Resource Class (E = external, I = internal)*	*Resource consume specs. (qty required)*	*Enabled Resource Cost Unit*	*Enabled Resource Cost of resource required*
autoclave_cure_L_edge	autocl_2650_sq_ft (I)	I, non-period	1 unit	$ 66 / unit	$ 66
	wip_L_edge_S004 (I)	I, non-period	4 units	$ 895 / unit	$ 3,580
	autoclave_hydro (E)	E, non-period	1,000 kwh/hr	$ 0. 05 / kwh	$ 300
	autocl_liq_nitrogen (E)	E, non-period	100 cu m/ hr	$ 0.10 / cu.m	$ 60
[Internal Activity]	autocl_depr (E)	E, non-period	1 hr / hr	$53,400/1780 hrs = $30 / hr	$ 180
	autoclave_eq (E)	E, non-period	1 unit	$ 0	$ 0
	autocl_686_oper (E)	E, non-period	1 unit	$ 0	$ 0
	autocl_686op_wage (E)	E, period	1 hr / hr	$ 25 / hr	$ 150
	autocl_686op_uben (E)	E, period	1 hr / hr	$ 8 / hr	$ 48
				TOTAL COST FOR 1 UNIT OF CURED_L_EDGE	$ 4,384
	Cost Order produced ...cured_L_edge (1 unit)		ABC per unit = $ 4384		

Given that the knowledge base of the deHavilland activities are input in terms of:-

- start and end time points of states associated with the activities as was indicated in section 6.4.1,
- the corresponding resource cost units of external resources and the resource cost units of internal resources as were deduced,

one may finally inquire about the cost of the cost order, cured_L_edge, for activity, autoclave_cure_L_edge. Given that all activity instances are completed, for example, by time point 19 to produce 1 unit of cured_L_edges, the form of the prolog query for cured_L_edge and the corresponding solution output up to the closest time point (in this case, time point 20) beyond completion time point 19 is shown in Figure 29 and the explanation that follows.

FIGURE 29 Prolog Query and Solution re: cpo for Cured Leading Edge Order



Prolog Query:- holdsT(project_order_costs (cured_L_edge, C), 20).

Solution: holdsT(project_order_costs (cured_L_edge, [0 [1], 4384 [2], 0 [3], 0 [4], 4384 [5]], 20)

where, for the cost order, cured_L_edge, upto time point 20 the activity costs are:-

[1] = dormant activity costs are $ 0

[2] = executed activity costs are $ 4,384

[3] = suspended activity costs are $ 0

[4] = reExecuted activity costs are $ 0

[5] = activity cost of cost order (1 unit) is $ 4,384 [= $ 0 + $ 4, 384 + $ 0 + $ 4,384]

6.7 Summary of Solutions to Competency Queries

All solutions are based upon a forensic ABC analysis of the cost data for the 1995 year.

1. **Competency Query:-** What should the company consider as being the 1995 ABC cost per cured leading edge?

Solution:- The 1995 ABC cost per cured leading edge is approximately $4,384.

2. **Competency Query:-** *Based upon an average autoclave activity instance, what should be the $ amount that the company allocate towards factory space depreciation associated with the activity?*

Solution: - The factory space depreciation cost is $18 per average autoclave activity instance.

3. **Competency Query:-** How many dollars of property tax expenditures should be allocated towards the autoclave activity instance producing cured leading edges?

Solution:- $12 for property taxes should be allocated towards the autoclave activity instance.

4. **Competency Query:-** What should be the overhead costs associated with the 2,650 square feet of space utilized by the autoclave activity producing leading edges?

Solution:- The cost of the 2,650 square feet of factory space utilized per autoclave activity instance is $66. This cost includes the cost of factory space depreciation ($18), property taxes ($12), factory insurances ($12), heat ($18) and hydro (6).

6.8 Concluding Remarks

The deductive solutions to queries as shown in this chapter have enabled us to conclude that a forensic cost analysis with ABC for cured leading edges indicates a cost of $4,384 per unit of cured_leading_edge.

First, the forensic cost analysis for deHavilland is made possible through the Principle of Resource Probing and the formalization of traditional notions of overheads within the

Theory of Resource Cost Units to achieve the Enveloped Activity Based Enterprise Model (EABEM) for deHavilland. Secondly, having populated the model with known resource cost units for the frontier activities and activity time completions to simulate historical time events, we were able to deduce the resource cost units for the internal resources. Finally, through the Cost Ontology in TOVE and the Micro-theory for ABC, solutions to some informal competency queries and the cost of a cost order are deduced in an unambiguous and consistent manner through formalization of cost point of resources (cpr), cost point of activities (cpa) and cost point of order (cpo).

CHAPTER 7 Enterprise Modelling and ABC at BHP

7.1 Introduction

This document discusses a prototype enterprise model and a cost analysis with ABC that spans the conventional notions of "direct and overhead (or indirect) activities" for BHP Steel (Broken Hill Proprietary, Western Port, Melbourne, Australia). This chapter, though based upon the same theories and Cost Advisor, is presented with a view to bringing out practical highlights of this research, other than the those already stated in the deHavilland study. However, similar to the deHavilland study, the chapter centres around the functionality of the Cost Advisor in providing solutions to the competency questions put forth by BHP Steel.

From a practical perspective, highlights of this study with reference to representing and reasoning about costs with enterprise models and ABC are:-

1. The Principle of Resource Probing facilitates the cost review of processes, the re-engineering of processes if necessary, and the "make or buy" decision for an internal resource of an enterprise. Let us examine this. By definition, an internal resource must have some "process" that produces it in-house. The Principle of

Resource Probing applied to an internal resource converges upon its "commit_envelope", i.e., the aggregate activity with sub-activities that lead to the production of the internal resource. In other words, knowing the commit_envelope of an internal resource, provides the company the "production process" for the internal resource. Therefore, all significant company activities, starting from a frontier activity up to the production of the probed resource are traceable, and can be reviewed from a cost and processing perspective. Therefore, the cost of the internal resource (or cost object) can be obtained inclusive of the direct, indirect and overhead costs borne by the enterprise as discussed in the deHavilland study. First, knowing the "true cost" of an internal resource, the company is in a better position to decide upon the "make or buy" policy for the internal resource. Second, knowing the costs and significant activities of production processes for internal resources, the company is in a better position to re-engineer processes as necessary. (Figure 36 shows the *sequential trace* of activities, identified in text form, in the commit envelope due to the Principle of Resource Probing being applied to an internal resource, rolling_slab1. Figure 37 graphically displays the commit envelope of the internal resource. The costs of each of these activities are obtained in the same manner as in the case with the deHavilland study).

2. The axioms encapsulated within the Cost Advisor provide the deductive reasoning capabilities to solve fairly complex cost related questions under what-if scenarios. (The solving of competency question 5 for this chapter illustrates the point.)

7.2 Brief Description about the BHP Study

The BHP implementation reinforces the applicability of ontologies, principles and theories developed in this research towards achieving ABC analysis through deductive reasoning. It describes the modelling and ABC analysis for a BHP product called coiled blackform.

The theories applied towards achieving the BHP activity based enterprise model implementation are:-

- the Principle of Resource Probing,

- the Commit Envelope of an Internal Resource,
- the establishment of an EABEM (Enveloped Activity Based Enterprise Model).

The reasoning and computations of ABC for the BHP product, coiled blackform, have been achieved through the application of the Cost Ontology and Micro-theory for ABC to the knowledge base of the enveloped BHP activity based model.

The knowledge base of EABEM for BHP was developed through consultations with BHP and EIL (U of T) personnel who have spent considerable time with BHP in Melbourne.

The study dwells around the process view for the manufacture of coiled blackform. It provides an insight into the object oriented representation of the enterprise model itself and the rationale of the model structure towards seeking solutions to queries from an ABC perspective. The study deals with the cost computations that conclude with the ABC cost for a unit of coiled blackform to include direct, indirect and overhead costs. The summary of cost computations have been based on simulated solutions to BHP competency queries. The solutions were obtained through OAK, ROCK and Prolog axioms that facilitate deductive reasoning about resources, activities and costs in the BHP knowledge base. The knowledge base was formulated with historical data about costs for external resources and activity event time data for activity instances. Cost figures have been altered wherever necessary to maintain confidentiality.

7.3 An Overview of Coiled Blackform Production at BHP

Each BHP product has a Manufacturing Route that consists of the manufacturing processes through which some form of raw material product must be processed through in order for some final product to be produced.

For purposes of prototyping the BHP model, the blackform product manufacturing at Western Port (wp) has the manufacturing route that consists of 2 major processes:-

1. shipping (sh) raw ingot slabs from facility or production unit #230 to the Hot Strip Mill facility or production unit #240, and

2. hot strip milling (hsm) at production unit #240.

The shipping activity is carried out in a rented yard space designated as yard_1. The yard_1 operator uses material handling equipment to transport each raw ingot slab on hand at yard_1 from production unit #230 to #240 for hot strip milling. The shipping or transportation of raw ingot slab from #230 to #240 takes approximately 3 hours for 96 slabs which are required to be oven-heated at #240 to fulfill one shift requirement for hot strip milling at #240.

The hot strip milling process at #240 consists of the sequential processes or sub-activities of oven-heating, roughing, rolling and coiling respectively. At the oven-heating stage, the raw ingot slabs are heated in batches for about 3 hours. The batch size for oven-heating is usually 96 units which sustains average production of coiled blackform for an 8 hour shift given that 1 unit of coiled blackform is produced every 5 minutes approximately. Each oven-heated slab is then passed on to the roughing stage where the slab is flattened or roughed by the roughing equipment for about 1.875 minutes per slab. The semi-flattened or roughened slab, then passes on to the rolling phase. The rolling phase consists of passing each roughened slab between rollers of gigantic rolling equipment. Each roughened slab is rolled for about 1.875 minutes between rollers to produce a much flattened and elongated rolled slab. Each rolled slab is then drawn into the coiling equipment in order to be coiled into a unit of coiled blackform. The coiling phase takes approximately 2.5 minutes to produce a unit of coiled blackform.

The oven capacity, production and shipping expediencies dictate that a batch size of 96 units be maintained as a process constraint.

From a simulated model perspective,

- an activity instance of shipping from #230 to #240 is completed in about 3 hours for 96 units of raw slabs,
- an activity instance of the oven-heating process is completed in about 3 hours for a batch of 96 raw slabs,

- an activity instance of the roughing process is completed in about 3 hours to produce 96 units of roughed slabs,
- an activity instance of the rolling process is completed in about 3 hours to produce 96 units of rolled slabs,
- an activity instance of the coiling process is completed in about 4 hours to produce 96 units of coiled blackform.

7.4 Competency Queries to be Solved

BHP was interested in the following competency queries to be solved with regard to their product - coiled blackform. Some of the BHP queries were retrospective in nature, while others involved looking into future what-if scenarios based upon some projected plans. These questions are stated in natural language as follows:-

1. What was the 1996 cost of shipping 96 units of raw slabs from #230 to #240 for oven-heating?
2. What were the 1996 costs of overheads for shipping 96 units of raw slabs from #230 to #240 for oven-heating?
3. What would be the 1997 cost of rolled slabs required at the coiling phase to produce 96 units of coiled blackform per shift if scheduled production hours is to be 3,000 hours for 1997?
4. Estimate gross profits before taxes on forecasted 1997 sales revenues of $10 million from coiled blackform units produced during 3,000 hours of scheduled production for 1997?

7.5 Use of the Principle of Resource Probing

A populated model that spanned 15 classes of activities and 60 instances of activities was created in ROCK for the TOVE Testbed. The algorithm for the Principle of Resource

Probing was applied to probe the end product or the cost object, coiled blackform (identified as coil_blkfrm in the knowledge base). In the trace of activities due to this probe, each activity class was tested for its significance, whether it was a frontier activity or internal activity, its required resources, and its cost orders (or cost objects). Resources were tested for significance and whether they were external or internal resources. Internal resources were in turn probed until frontier activities were checked out in the populated model.

The summarized result of the Principle of Resource Probing for coil_blkfrm is presented in Table 15.

TABLE 15Results of Principle of Resource Probing for BHP

Activity Class Name	*Cost Order of Activity*	*Resources Reqd (E = external; I = internal)*	*Identified Activity that produces Internal Resource*
wp_hsm_240_coiling	coil_blkfrm	coiling_200th_space (I)	provide_coiling_200th_space [1]
		coiling_eq (E)	
		coiling_oper (E)	
		coiling_mgr (E)	
		coiling_hydro (E)	
[Internal Activity]		coiling_eq_depr (E)	
		coiling_opwages (E)	
		coiling_opben (E)	
		coiling_mgrwages (E)	
		rolling_slab (I)	wp_hsm_240_rolling [2]
provide_coiling_200th_space [1]	coiling_200th_space	coilspace_depr_recd (E)	
		coiltax_recd (E)	
[Frontier Activity]		coilinsur_recd (E)	
		coilspace_heat (E)	
		coilspace_hydro (E)	
wp_hsm_240_rolling [2]	rolling_slab	rolling_200th_space (I)	provide_rolling_200th_space [3]
		rolling_eq (E)	
		rolling_oper (E)	
		rolling_mgr (E)	

Activity Class Name	*Cost Order of Activity*	*Resources Reqd (E = external; I = internal)*	*Identified Activity that produces Internal Resource*
		rolling_hydro (E)	
[Internal Activity]		rolling_eq_depr (E)	
		rolling_opwages (E)	
		rolling_opben (E)	
		rolling_mgrwages (E)	
		rolling_mgrben (E)	
		roughing_slab (I)	wp_hsm_240_roughing [4]
provide_rolling_200th_space [3]	rolling_200th_space	rollspace_depr_recd (E)	
		rolltax_recd (E)	
[Frontier Activity]		rollinsur_recd (E)	
		rollspace_heat (E)	
		rollspace_hydro (E)	
wp_hsm_240_roughing [4]	roughing_slab	roughing_200th_spac e (I)	provide_roughing_200th_space[5]
		roughing_eq (E)	
		roughing_oper (E)	
		roughing_mgr (E)	
		roughing_hydro (E)	
[Internal Activity]		roughing_eq_depr (E)	
		roughing_opwages (E)	
		roughing_opben (E)	
		roughing_mgrwages (E)	
		roughing_mgrben (E)	
		ovenhtg_1_slab96 (I)	wp_hsm_240_ovenhtg [6]
provide_roughing_200th_space[5]	roughing_200th_spa ce	rghspace_depr_recd (E)	
		rghtax_recd (E)	
[Frontier Activity]		rghinsur_recd (E)	
		rghspace_heat (E)	
		rghspace_hydro (E)	
wp_hsm_240_ovenhtg [6]	ovenhtg_1_slab96	ovenhtg_200th_space (I)	provide_ovenhtg_200th_space [7]
		ovenhtg_eq (E)	
		ovenhtg_oper (E)	
		ovenhtg_mgr (E)	
		ovenhtg_hydro (E)	

Activity Class Name	*Cost Order of Activity*	*Resources Reqd (E = external; I = internal)*	*Identified Activity that produces Internal Resource*
[Internal Activity]		ovenhtg_eq_depr (E)	
		ovenhtg_opwages (E)	
		ovenhtg_opben (E)	
		ovenhtg_mgrwages (E)	
		ovenhtg_mgrben (E)	
		wp_sh_230_slab (I)	wp_sh_230 [8]
provide_ovenhtg_200th_space [7]	ovenhtg_200th_space	ovspace_depr_recd (E)	
		ovtax_recd (E)	
[Frontier Activity]		ovinsur_recd (E)	
		ovspace_heat (E)	
		ovspace_hydro (E)	
wp_sh_230 [8]	wp_sh_230_slab	yard1_space (E)	
		yard1_oper (E)	
		yard1_eq (E)	
		yard1_rent (E)	
		yard1_insur (E)	
[Frontier Activity]		yard1_hydro (E)	
		yard1_opwages (E)	
		yard1_opben (E)	
		yard1_eq_depr (E)	
		yard1_eq_fuel (E)	
		yard1_rawslab (E)	

7.6 Modelling BHP Processes in ROCK

The BHP product (manufacturing) route is considered an aggregate class activity, route_aggr_act, to form a subclass of the significant BHP activities. The blackform product manufacturing at Western Port has sub-activities shipping and hot strip milling, designated as wp_sh_230 and wp_hsm_240 respectively. Each of these sub-activities are activities of production units #230 and #240 respectively. The hot strip milling process has sub-activities oven-heating, roughing, rolling and coiling. A sample ROCK model representation of the various class activities and their sub-activities is shown in Figure 30.

FIGURE 30 Sample ROCK Representation for BHP

```
class_frame route_aggr_act
{
subclassOf: bhp_significant_act;
}
class_frame production_unit_act
{
subclassOf: bhp_significant_act;
}
class_frame wp_blkform_route
{
subclassOf: route_aggr_act;
relations:
        has_subactivity = {wp_sh_230, wp_hsm_240 };
}
class_frame wp_sh_230
{
subclassOf: production_unit_act;
}
class_frame wp_hsm_240
{
subclassOf: production_unit_act;
relations:
        has_act_cost_order = { coil_blkform1 };
        has_subactivity = { wp_hsm_240_ovenhtg,
        wp_hsm_240_roughing,
        wp_hsm_240_rolling,
        wp_hsm_240_coiling };
}
```

In Figure 31, each of the sub-activities oven-heating, roughing, rolling and coiling of hot strip milling at #240 is in turn modelled in ROCK as a class_frame activity that is a sub-class of BHP significant activities.

FIGURE 31 ROCK Frames re: Sub-activities of Hot Strip Milling at #240

```
class_frame wp_hsm_240_ovenhtg
{
subclassOf: bhp_significant_act ;
relations:
        subactivity_of = {wp_hsm_240 };
}
class_frame wp_hsm_240_roughing
{
subclassOf: bhp_significant_act ;
relations:
        subactivity_of = {wp_hsm_240 };
}
class_frame wp_hsm_240_rolling
{
subclassOf: bhp_significant_act ;
relations:
        subactivity_of = {wp_hsm_240 };
}
class_frame wp_hsm_240_coiling
{
subclassOf: bhp_significant_act ;
relations:
        subactivity_of = {wp_hsm_240 };
}
```

Each of the class activities of shipping at #230 and the sub-class activities of hot strip milling at #240 are instantiated with a view to querying the ROCK model to obtain cost related solutions about resources, activities and cost orders of activities consistent with the Cost Ontology of Ch. 4 and the Formalization of Overheads in Ch. 5. Figure 32 illustrates the activity instance for shipping at #230 and its enabling states in ROCK to make possible the capture of direct and overhead costs with the cost axioms. Similarly, Figure 33 illustrates the ROCK representation for an activity instance at production unit #240.

FIGURE 32 ROCK Frames re: Instantiating and Enabling Shipping Activity at Production Unit #230

```
named_instance_frame wp_sh_230_1
{
instanceOf: wp_sh_230 ;
relations:
        enabled_by = {es_wp_sh_230_1};
        causes = {cs_wp_sh_230_1};
        has_act_cost_order = {wp_sh_230_slab};
        has_act_cost_object = {wp_sh_230_1_act_cost};
attributes:
        batch_time = 3;
}
named_instance_frame es_wp_sh_230_1
{
instanceOf: bhp_state_conjunct ;
relations:
        enables = {wp_sh_230_1};
        conjuncts = { use_yard1_space, use_yard1_oper, use_yard1_eq,
        cons_yard1_rent, cons_yard1_insur, cons_yard1_hydro,
        cons_yard1_opwages, cons_yard1_opben,
        cons_yard1_eq_depr, cons_yard1_eq_fuel,
        cons_yard1_rawslab};
}
```

The conjunctive enabling states of the shipping activity instance of Figure 32 associates direct material cost entities such as rawslab and equipment fuel (cons_eq_fuel), and direct labour cost entities such as operator wages (opwages). Overhead cost entities associated with the activity are rent, insurance, hydro, operator supplementary benefits (opben) and shipping equipment depreciation (eq_depr).

The conjunctive enabling states of the roughing activity instance of Figure 33 have been represented with a view to capturing the direct material cost entity borne by each oven-heated slab (ovenhtg_1_slab96) and the notional overheads related to the usage roughing_space, equipment, operator, supervisor/manager and the consumption of equipment power supply (hydro), equipment depreciation (eq_depr), operator wages(opwages), operator supplementary benefits (opben), supervisor/manager wages (mgrwages), and manager supplementary benefits (mgrben). Each instance of roughing activity produces a unit of roughed slab (roughing_slab1). In ABC terminology, roughing_slab1 is the cost order or cost object of the roughing activity.

FIGURE 33 ROCK Frames re: Instantiating and Enabling Roughing Activity at Production Unit #240 (HSM)

```
named_instance_frame wp_hsm_240_roughing_1
{
instanceOf: wp_hsm_240_roughing ;
relations:
        enabled_by = {es_240_roughing_1};
        causes = {cs_240_roughing_1 ;
        has_act_cost_order = {roughing_slab1};
        has_act_cost_object = {roughing_1_act_cost};
attributes:
        batch_time = 3;
}
named_instance_frame es_240_roughing_1
instanceOf: bhp_state_conjunct;
relations:
        enables = {wp_hsm_240_roughing_1};
        conjuncts = {use_roughing_space, use_roughing_eq, use_roughing_oper,
        use_roughing_mgr, cons_roughing_hydro,
        cons_roughing_eq_depr, cons_roughing_opwages,
        cons_roughing_opben, cons_roughing_mgrwages,
        cons_roughing_mgrben, cons_ovenhtg_1_slab96};
}
```

The roughing activity is considered to require an overall factory space of about 200,000 square feet. For purposes of model building, it has been estimated that each of oven-heating, roughing, rolling and coiling phases for coiled blackform production at #240 uses approximately 200,000 square feet each respectively.

Frame definitions shown in Figure 34 for modelling factory overheads associated with factory space usage required for the roughing activity are based on the consumption of fixed factory space depreciation, fixed property tax, fixed insurance, hydro and heat energy. Note the modelling of the nonactivity cost fluent, bldg_depreciation(tps, tc_ext, tt_act, r) highlighted in Figure 34. Axiom (FOL 97) deduces the unit cost for resource, rghspace_depr_recd, to be $10 / hr in Table 16.

FIGURE 34 ROCK Frames re: Factory Space Overheads towards Roughing Activity

```
named_instance_frame provide_roughing_200th_space
{
instanceOf: bhp_support1_acts ;
relations:
        enabled_by = {es_provide_roughing_200th_space};
        causes = {cs_provide_roughing_200th_space};
        has_act_cost_order = {roughing_200th_space};
        has_act_cost_object = {provide_roughing_200th_space_act_cost};
}
named_instance_frame es_provide_roughing_200th_space
{
instanceOf: bhp_state_conjunct;
relations:
        enables = {provide_roughing_200th_space};
        conjuncts = {cons_rghspace_depr_recd, cons_rghtax_recd,
        cons_rghinsur_recd, cons_rghspace_heat,
        cons_rghspace_hydro };
}
named_instance_frame cons_rghspace_depr_recd
{
instanceOf: cons_bldg_depreciation;
relations:
        has_nonactivity_cost_fluent = {bldg_depr_cost_rghspace};
        consumes = {rghspace_depr_recd};
attributes:
        quantity = 1;
}
named_instance_frame bldg_depr_cost_rghspace
{
instanceOf: nonactivity_cost_fluent;
relations: nonactivity_cost_fluent_of = {cons_rghspace_depr_recd};
attributes: {tps = 1, tc_ext = 30000, tt_act = 3000, rname = rghspace_
depr_recd}; }
```

[refer to Table 2]

Similar considerations regarding factory space usage for oven-heating, rolling and coiling enable us to deduce the factory overheads associated with each activity instance and activity class that may be called for towards the production of the cost order, coiled blackform.

The sample ROCK class_frames of Figures 31-34 compose the knowledge base of EABEM. This representation of the enterprise model provides the infrastructure for deductive reasoning of solutions to competency queries through the application of the axioms formalized in this research.

7.7 Deducing Solutions for Cost Queries about Coiled Blackform

The queries put forth by BHP are by no means exhaustive and complete. However, solutions to the BHP queries demonstrates the applicability of the contributions towards enterprise modelling and activity based costing developed within this research. Solutions to the BHP queries provide some insight into how this research can assist companies towards gaining an advantage in strategic cost management.

7.7.1 Significant Activities

The significant activities in the BHP enterprise activity model that are required to produce coiled blackform are:-

- wp_sh_230 .... shipping or material transport of raw slab from #230 to #240,
- wp_hsm_240 .... hot strip milling that has sub-processes or sub-activities,
- wp_hsm_240_ovenhtg .... first sub-activity phase of hot strip milling at #240,
- wp_hsm_240_roughing .... second sub-activity phase of hot strip milling at #240,
- wp_hsm_240_rolling .... third sub-activity phase of hot strip milling at #240,

- wp_hsm_240_coiling .... final sub-activity phase of hot strip milling at #240 towards coiled blackform at #240,
- provide_ovenhtg_200th_space ..... 200, 000 square feet required for ovenheating,
- provide_roughing_200th_space .... 200, 000 square feet required for roughing,
- provide_rolling_200th_space ...... 200, 000 square feet required for rolling,
- provide_coiling_200th_space .... 200, 000 square feet required for coiling.

7.7.2 Solution to Question 1

Question 1: What was the 1996 cost of shipping 96 units of raw slab from #230 to #240 for oven-heating?

Table 12 summarizes the relevant data modelled for the shipping activity at #230.

TABLE 16 Activity Instance of Class Activity wp_sh_230 [Activity time = 3 hours]

Activity Name	*Total Actual Activity Time for 1996 instances (T)*	*External (E) & Internal (I) Resources Required*	*Total 1996 Costs of Resources (C)*	*Output per instance*	*Enabled Resource Cost Units * (C) / (T)*
wp_sh_230_1 [Frontier Activity]	1, 500 hours	[P = period cost res] [NP = nonperiod cost res]		96 raw-slabs	
		yard1_space (E) (NP)			----
		yard1_oper (E) (NP)			----
		yard1_eq (E) (NP)			----
		yard1_rent (E) (P)	$ 150,000		$ 100 per hour
		yard1_insur (E) (P)	$ 75,000		$ 50 per hour
		yard1_hydro (E) (NP)			$ 1 per kwh [10 kwh consumed per hour]
		yard1_opwages (E) (P)	$ 45,000		$ 30 per hour
		yard1_opben (E) (P)	$ 15,000		$ 10 per hour
		yard1_eq_depr (E) (P)	$ 30,000		$ 20 per hour

Activity Name	*Total Actual Activity Time for 1996 instances (T)*	*External (E) & Internal (I) Resources Required*	*Total 1996 Costs of Resources (C)*	*Output per instance*	*Enabled Resource Cost Units * (C) / (T)*
		yard1_eq_fuel (E) (NP)			$1 per litre [10 litres consumed per hour]
		yard1_rawslab (E) (NP)			$ 200 per slab; $19,200 for batch of 96 slabs

Assuming that the activity event time begins at time point 1 and is completed at time point 4, we may query the model to obtain a solutions at time point 5.

- Comment: The above question is equivalent to asking what is the cost point of activity (cpa) wp_sh_230_1 at time point 5, which is the closest integer time point for the activity being completed at time point 4.
- Comment: Noting that the cost order of activity wp_sh_230_1 is wp_sh_230_slab, we may deduce the solution to the question, by querying the model for the cost point of order (cpo) at time point 5 for the cost order wp_sh_230_slab.

  Prolog Query:- holdsT(project_act_costs (wp_sh_230_1, C), 5).

  Solution: holdsT(project_act_costs (wp_sh_230_1, [0[1], 19890[2], 0[3], 0[4], 19890[5]]), 5)

  Prolog Query:- holdsT(project_order_costs(wp_sh_230_slab, C, 5).

  Solution: holdsT(project_order_costs (wp_sh_230_slab, [0[1], 19890[2], 0[3], 0[4], 19890[5]]), 5)

[1] = dormant activity cost; [2] = execute activity cost; [3] = suspend activity cost; [4] = reExecute activity cost; [5] = [1] + [2] + [3] + [4] = completed activity cost

Solution: In 1996, it cost $19, 890 for shipping 96 units of raw slabs from #230 to #240 for oven-heating (refer Table 16).

7.7.3 Solution to Question 2

Question 2:- What were the 1996 costs of overheads for shipping 96 units of raw slabs from #230 to #240 for oven-heating?

- Comment: [Refer to Table 2]. Given that the notions of overheads include yard1 rent, equipment depreciation, insurance, hydro, and operator supplementary benefits, we can query the cost point of resource (cpr) for each of the resources yard1_rent, yard1_eq_depr, yard1_insur, yard1_opben.

Assuming that the activity event time begins at time point 1 and is completed at time point 4, we may query the model to obtain a solution at time point 5 as was done in the previous section.

> Prolog Query:- holdsT (project_res_costs(cons_yard1_rent, yard1_rent, wp_sh_230_1, C), 5).
>
> Solution: holdsT(project_res_costs (cons_yard1_rent, yard1_rent, wp_sh_230_1, [0,300, 0, 0, 300]), 5)

Solution: The overhead cost due to yard1_ rent is $300 from solution to the prolog query. In like manner, overhead costs due to insurance, hydro, equipment depreciation, and operator supplementary benefits are $150, $30, $60, $30 respectively. Total overhead cost towards shipping 96 slabs from #230 to #240 is $570 (= $300 + $150 + $30 + $60 + $30).

7.7.4 Solution to Question 3

Question 3:- What would be the 1997 cost of rolled slabs required at the coiling phase to produce 96 units of coiled blackform per shift if scheduled production hours is to be 3,000 hours for 1997?

- Comment: One must first ascertain if the resource, rolling slab, required at coiling is an external or internal resource. If the enterprise model confirms that rolling slab is a frontier or an external resource, then the commit resource cost unit of rolling slab must be a given and would be equivalent to the purchase cost of the resource. This would be a trivial situation. However, if rolling slab is an internal (or non-frontier) resource, the commit resource cost unit for rolling slab must be deduced by the cost of activities that contribute to the "commit envelope" of rolling slab.

Prolog Query:- external_res(rolling_slab1). *[To check if resource is external]*

Solution: No solution found.

[The solution indicates that the resource is not external, and the visual display confirms that the resource is not external as it is a resource that is produced by a terminal produce state as shown in figure 35].

FIGURE 35 Output Solution to Query: Is rolling_slab1 an external resource in BHP model?

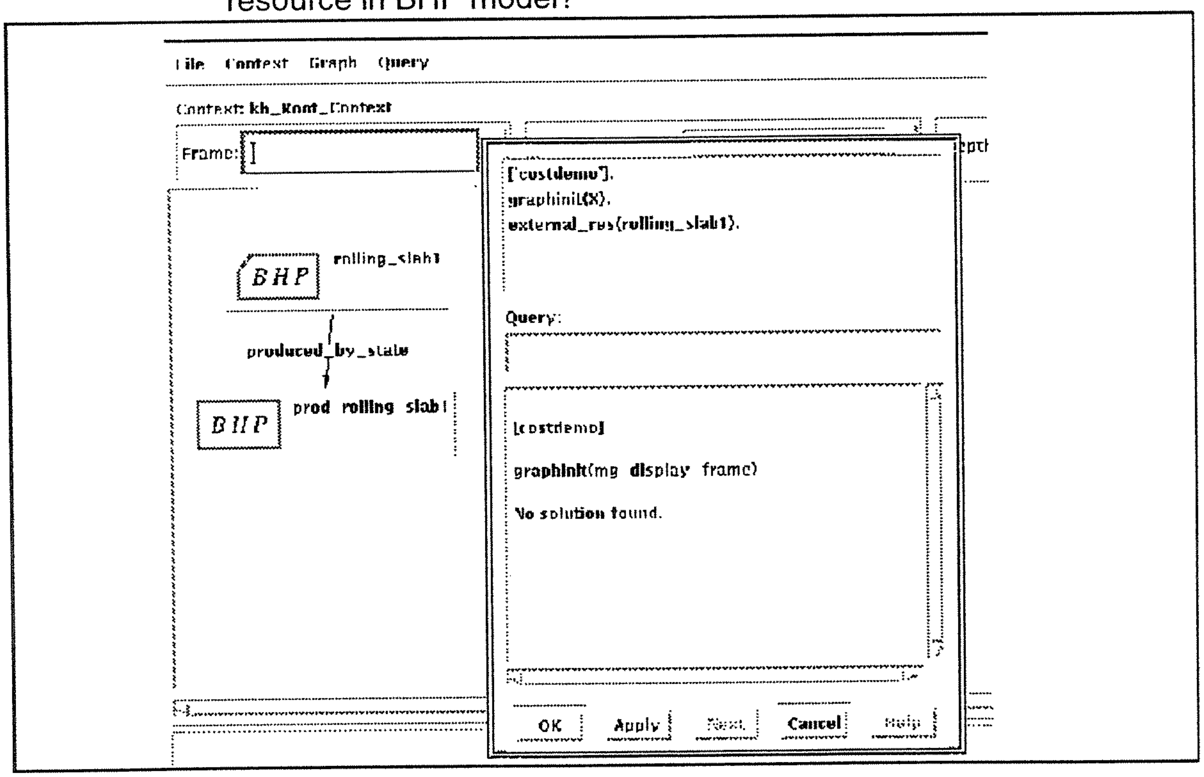


FIGURE 36 Text Solution re: Commit Envelope of rolling_slab1

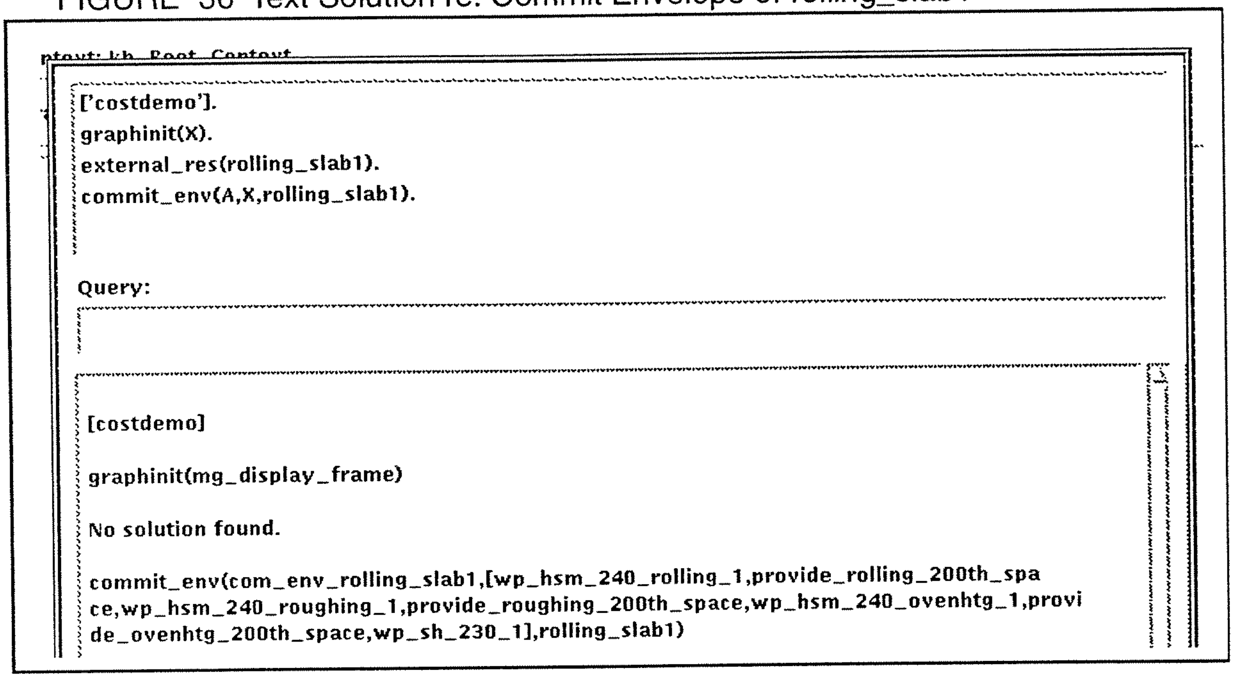
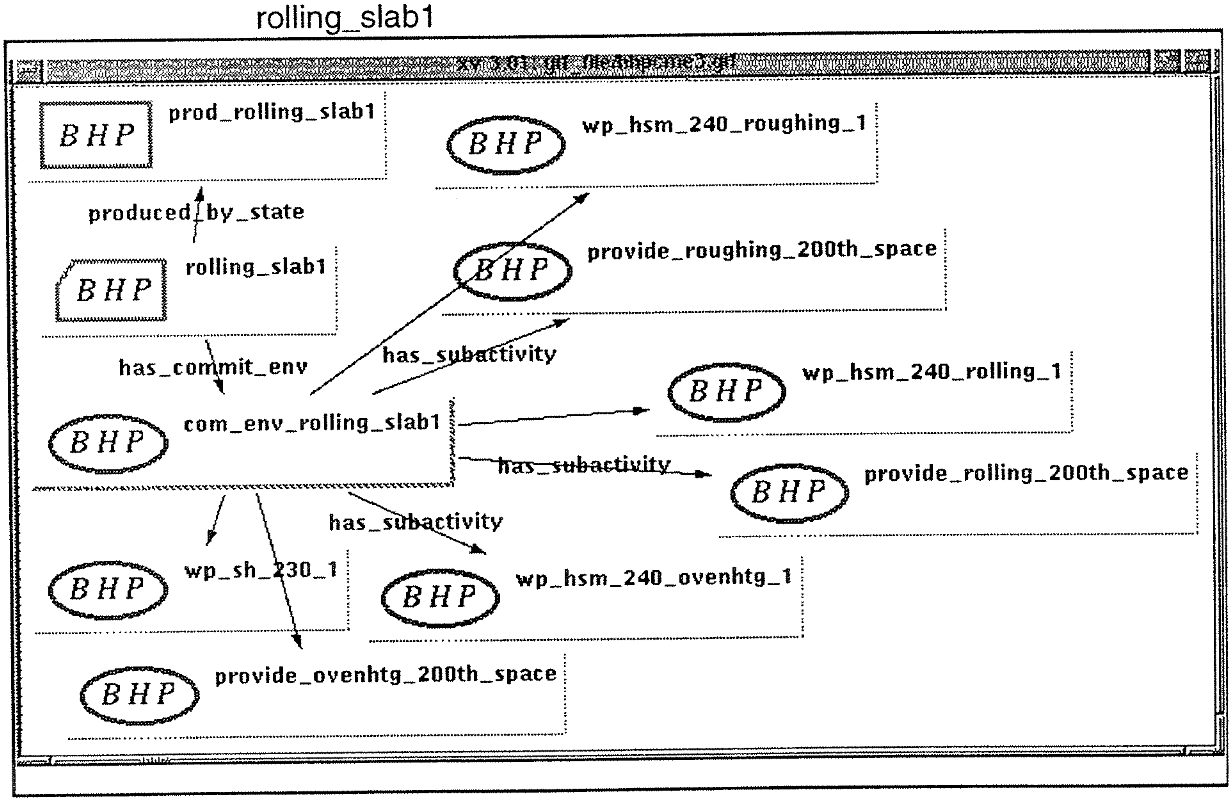


FIGURE 37 Visual Solution re: Sub-actions of Commit Envelope of rolling_slab1



Prolog Query:- commit_env (A, X, rolling_slab1).

Solution: [Figure 36 provides the text solution that indicates that the aggregate activity, com_env_rolling_slab1, is the commit envelope of rolling_slab1 and lists the sub-activities of the commit envelope. Figure 37 provides the simultaneously produced visual display of the text solution of the activities that are the sub-activities of the com_env_rolling_slab1. This information together with costs of activities would prove invaluable to enterprises for "make-buy" decisions of resources, and the re-engineering of processes, if necessary].

To deduce the commit resource cost unit of rolling slab, the cost of the sub-activities of the commit envelope for rolling slab must be deduced. The results of the cost of resources required by sub-activities of the commit envelope are summarized in Tables 17 - 23.

TABLE 17Cost of Resources Required: activity "wp_sh_230_1" .....(Time = 3 hrs)

Activity Name	*Resource*	*Resource Class (E = external; I = internal)*	*Resource consume/ usage specs. (qty q required)*	*Enabled Resource Cost Unit (v)*	*[1]Enabled Resource cost of resource required*
wp_sh_230_1	yard1_space	E, non-period	1 unit	$ 0	$ 0
	yard1_oper	E, non-period	1 unit	$ 0	$ 0
	yard1_eq	E, non-period	1 unit	$ 0	$ 0
	yard1_rent	E, period	1 hr / hr	*$100 / hr	$ 300
	yard1_insur	E, period	1 hr / hr	* $ 50 / hr	$ 150
[Frontier Activity]	yard1_hydro	E, non-period	10 kwh / hr	$ 1 / hr	$ 30
	yard1_opwages	E, period	1 hr / hr	$ 30 / hr	$ 90
	yard1_opben	E, period	1 hr / hr	$ 10 / hr	$ 30
	yard1_eq_depr	E, period	1 hr / hr	$ 20 / hr	$ 60
	yard1_eq_fuel	E, non-period	10 litres / hr	$ 1 / litre	$ 30
	yard1_rawslab	E, non-period	96 slabs	$ 200 / slab	$ 19,200
				TOTAL COST	$ 19,890 for 96 units
	Resource produced.......wp_sh_230_slab				

* Fixed costs for yard1 rent and yard1 insurance is projected to be $300, 000 and $150,000 respectively for 1997. Hence, resource cost units are $100/hour (= $300,000/3,000 hours) and $50/hour (= $150,000/3,000 hours) respectively, where 3,000 hours represents estimated actual activity or plant utilization. Note, as pointed out in the deHavilland study, higher plant utilization will lower the overhead cost component in the final product price.

The prolog query and solution form for the enabled resource cost of resource yard1_insur for the activity wp_sh_230_1 is shown in Figure 38. The solution of Figure 38 indicates that the enabled resource cost of yard1_insurance is $150 for the activity instance.

FIGURE 38 Prolog Query & Solution re: enabled_res_cost of yard1_insur

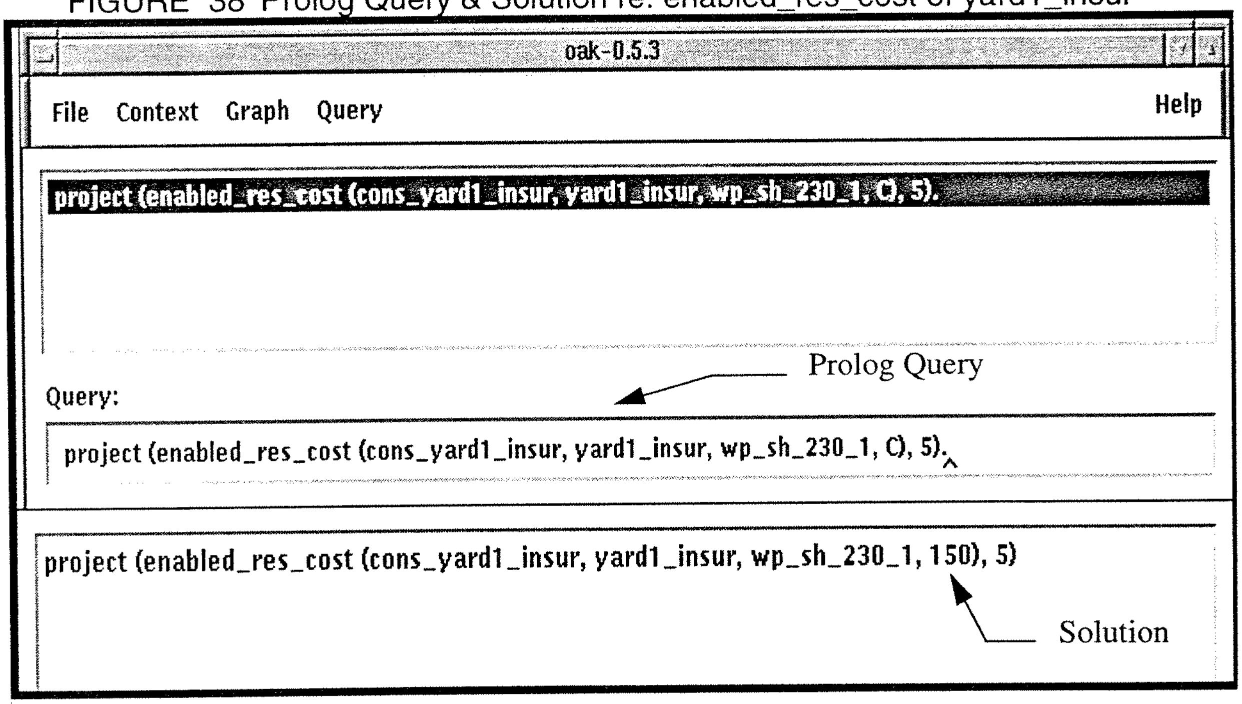


TABLE 18Cost of Resources Required for Instance of Activity "provide_ovenhtg_200th_space" .....(Time = 3 hrs)

Activity Name	*Resource*	*Resource Class (E = external, I = internal)*	*Resource consume/ usage specs. (qty q required)*	*Enabled Resource Cost Unit (v)*	*Enabled Resource cost of resource required*
provide_ovenhtg_200th_space	ovspace_depr_recd	E, period	1 hr / hr	* $ 10 / hr	$ 30
	ovtax_recd	E, period	1 hr / hr	$ 20 / hr	$ 60
[Frontier Activity]	ovinsur_recd	E, period	1 hr / hr	$ 30 / hr	$ 90
	ovspace_heat	E, non-period	10 cu. m / hr	$ 1 / cu. m	$ 30
	ovspace_hydro	E, non-period	10 Kwh / hr	$ 1 / Kwh	$ 30
				TOTAL COST	$ 240 / unit batch of 96
	Resource produced............1 unit of ovenhtg_200th_space				

* Projected 1997 annual depreciation for 200, 000 square feet of factory space is approximately $30,000 per year; forecasted oven utilization hours is 3,000 hours for the year. As explained in the deHavilland study, the enabled resource cost unit for ovspace_depr_recd would be $30,000 / 3,000 hrs = $10 / hr.

Similarly, projected 1997 annual apportioned property tax and insurance estimates are $60,000 and $90,000 respectively. Hence, enabled resource cost units for ovtax_recd and ovinsur_recd are $60,000 / 3,000 hrs = $ 20 / hour, and $90,000 / 3,000 hrs = $ 30 / hour respectively.

The computed enabled resource cost units for ovenheating space depreciation, property tax, insurance, heat and hydro have been considered applicable to the other activities of roughing, rolling and coiling of hot strip milling process at #240.

TABLE 19Cost of Resources Required for Instance of Activity "wp_hsm_240_ovenhtg"..... (Time = 3 hrs)

Activity Name	*Resource*	*Resource Class (E = external, I = internal)*	*Resource consume/ usage specs. (qty q required)*	*Enabled Resource Cost Unit (v)*	*Enabled Resource cost of resource required*
wp_hsm_240_ovenhtg	ovenhtg_200th_space	I , non-period	1 unit	$ 240 / unit	$ 240
	ovenhtg_eq	E, non-period	1 unit	$ 0 / unit	$ 0
	ovenhtg_oper	E, non-period	1 unit	$ 0 / unit	$ 0
	ovenhtg_mgr	E, non-period	1 unit	$ 0 / unit	$ 0
	ovenhtg_hydro	E, non-period	10,000 Kwh / hr	$ 0.25 / Kwh	$ 7,5 00
[Internal Activity]	ovenhtg_eq_depr	E, period	1 hr / hr	$ 100 / hr *	$ 300
	ovenhtg_opwages	E, period	1 hr / hr	$ 20 / hr **	$ 60
	ovenhtg_opben	E, period	1 hr / hr	$ 10 / hr **	$ 30
	ovenhtg_mgrwages	E, period	1 hr / hr	$ 40 / hr **	$ 120
	ovenhtg_mgrben	E, period	1 hr / hr	$ 20 / hr **	$ 60
	wp_sh_230_slab	I, non-period	batch of 96 slabs	$ 19, 890 for batch of 96	$ 19, 890
				TOTAL COST	$ 28, 200 for batch of 96
	Resource produced............ovenhtg_1_slab96				

* Projected 1997 annual depreciation of oven-heating equipment = $300,000; projected oven-heating equipment usage = 3,000 hrs; enabled resource cost unit = $300,000 / 3,000 hrs = $100 / hr.

** Note that these were given by BHP, and were expressed as a cost per hour. Factory operator benefits and management benefits were said to be 50% of the respective hourly cost

TABLE 20 Resource Costs: Activity "provide_roughing_200th_space" (Time = 3 hrs)

Activity Name	*Resource*	*Resource Class (E = external, I = internal)*	*Resource consume/ usage specs. (qty q required)*	*Enabled Resource Cost Unit (v)*	*Enabled Resource cost of resource required*
provide_roughing _200th_space	rghspace_depr_re cd	E, period	1 hr / hr	$ 10 / hr	$ 30
	rghtax_recd	E, period	1 hr / hr	$ 20 / hr	$ 60
	rghinsur_recd	E, period	1 hr / hr	$ 30 / hr	$ 90
[Frontier Activity]	rghspace_heat	E, non-period	10 cu. m / hr	$ 1 / cu. m	$ 30
	rghspace_hydro	E, non-period	10 Kwh / hr	$ 1 / Kwh	$ 30
				TOTAL COST	$ 240 / unit
	Resource produced............1 unit of roughing_200th_space				

TABLE 21 Resource Costs: Activity "wp_hsm_240_roughing"..... (Time = 3 hrs)

Activity Name	*Resource*	*Resource Class (E = external, I = internal)*	*Resource consume/ usage specs. (qty q required)*	*Enabled Resource Cost Unit (v)*	*Enabled Resource cost of resource required*
wp_hsm_240_rou ghing	roughing_200th_spa ce	I, non-period	1 unit	$ 240 / unit	$ 240
	roughing_eq	E, non-period	1 unit	$ 0 / unit	$ 0
	roughing_oper	E, non-period	1 unit	$ 0 / unit	$ 0
	roughing_mgr	E, non-period	1 unit	$ 0 / unit	$ 0
	roughing_hydro	E, non-period	2,000 Kwh / hr	$ 0.25 / Kwh	$ 1, 500
[Internal Activity]	roughing_eq_depr	E, period	1 hr / hr	$ 40 / hr	$ 120
	roughing_opwages	E, period	1 hr / hr	$ 20 / hr	$ 60
	roughing_opben	E, period	1 hr / hr	$ 10 / hr	$ 30
	roughing_mgrwages	E, period	1 hr / hr	$ 40 / hr	$ 120
	roughing_mgrben	E, period	1 hr / hr	$ 20 / hr	$ 60
	ovenhtg_1_slab96	I, non-period	batch of 96 slabs	$ 28,200 for batch of 96	$ 28, 200
				TOTAL COST	$ 30, 330 for batch of 96
	Resource produced..............roughing_slab96				

TABLE 22Resource Costs: Activity "provide_rolling_200th_space" (Time = 3 hrs)

Activity Name	*Resource*	*Resource Class (E = external, I = internal)*	*Resource consume/ usage specs. (qty q required)*	*Enabled Resource Cost Unit (v)*	*Enabled Resource cost of resource required*
provide_rolling_200th_space	rollspace_depr_recd	E, period	1 hr / hr	$ 10 / hr	$ 30
	rolltax_recd	E, period	1 hr / hr	$ 20 / hr	$ 60
	rollinsur_recd	E, period	1 hr / hr	$ 30 / hr	$ 90
[Frontier Activity]	rollspace_heat	E, non-period	10 cu. m / hr	$ 1 / cu. m	$ 30
	rollspace_hydro	E, non-period	10 Kwh / hr	$ 1 / Kwh	$ 30
				TOTAL COST	$ 240 / unit
	Resource produced...........1 unit of rolling_200th_space				

TABLE 23Resource Costs: Activity "wp_hsm_240_rolling"..... (Time = 3 hrs)

Activity Name	*Resource*	*Resource Class (E = external, I = internal)*	*Resource consume/ usage specs. (qty q required)*	*Enabled Resource Cost Unit (v)*	*Enabled Resource cost of resource required*
wp_hsm_240_rolling	rolling_200th_space	I, non-period	1 unit	$ 240 / unit	$ 240
	rolling_eq	E, non-period	1 unit	$ 0 / unit	$ 0
	rolling_oper	E, non-period	1 unit	$ 0 / unit	$ 0
	rolling_mgr	E, non-period	1 unit	$ 0 / unit	$ 0
	rolling_hydro	E, non-period	2,000 Kwh / hr	$ 0.25 / Kwh	$ 1, 500
[Internal Activity]	rolling_eq_depr	E, period	1 hr / hr	$ 50 / hr	$ 150
	rolling_opwages	E, period	1 hr / hr	$ 20 / hr	$ 60
	rolling_opben	E, period	1 hr / hr	$ 10 / hr	$ 30
	rolling_mgrwages	E, period	1 hr / hr	$ 40 / hr	$ 120
	rolling_mgrben	E, period	1 hr / hr	$ 20 / hr	$ 60
	roughing_1_slab96	I, non-period	batch of 96 slabs	$30, 330 for batch of 96	$ 30, 330
				TOTAL COST	$ 32, 490 for batch of 96
	Resource produced............... .....rolling_1_slab96				

Solution:- The 1997 commit resource cost unit would be $32,490 per batch of 96 rolled slabs that are required for the coiling phase of hot strip milling at #240. For 3,000 hours of production in 1997, there would be 1,000 activity instances for the rolling phase, given that each activity instance of wp_hsm_240_ rolling is completed in 3 hours.

Therefore, the total estimated 1997 costs of rolled slabs at the coiling phase is $32,490,000 to produce 96 units of coiled blackform per shift if actual production hours are to be 3,000 hours for 1997.

7.7.5 Solution to Question 4

Question 4: Estimate gross profits before taxes on forecasted 1997 sales revenues of $28 million from coiled blackform units produced during 3,000 hours of scheduled production for 1997?

The coiling phase of hot strip milling at #240 is the final phase in the production of coiled blackform. The coiling activity requires 200, 000 square feet of factory space as well as rolled slabs from the rolling phase of hot strip milling. A solution to this query is sought given that the production of all rolled slabs are committed and consumed by the coiling phase without any time delays.

In order to visually display the activity cluster, one may query the Oak browser as to what is the activity cluster instance for the Western Port, hot strip milling, facility #240, coiling activity. This is done as follows:-

> Prolog Query:- activity_cluster (wp_hsm_240_coiling_1).
>
> Solution: [The solution is visually displayed as shown in Figure 39].

If one desires the enabling states for the coiling activity, one may input the prolog query and obtain the text solution as shown in Figure 40.

FIGURE 39 Solution Display of Activity Instance of Coiling Phase at BHP

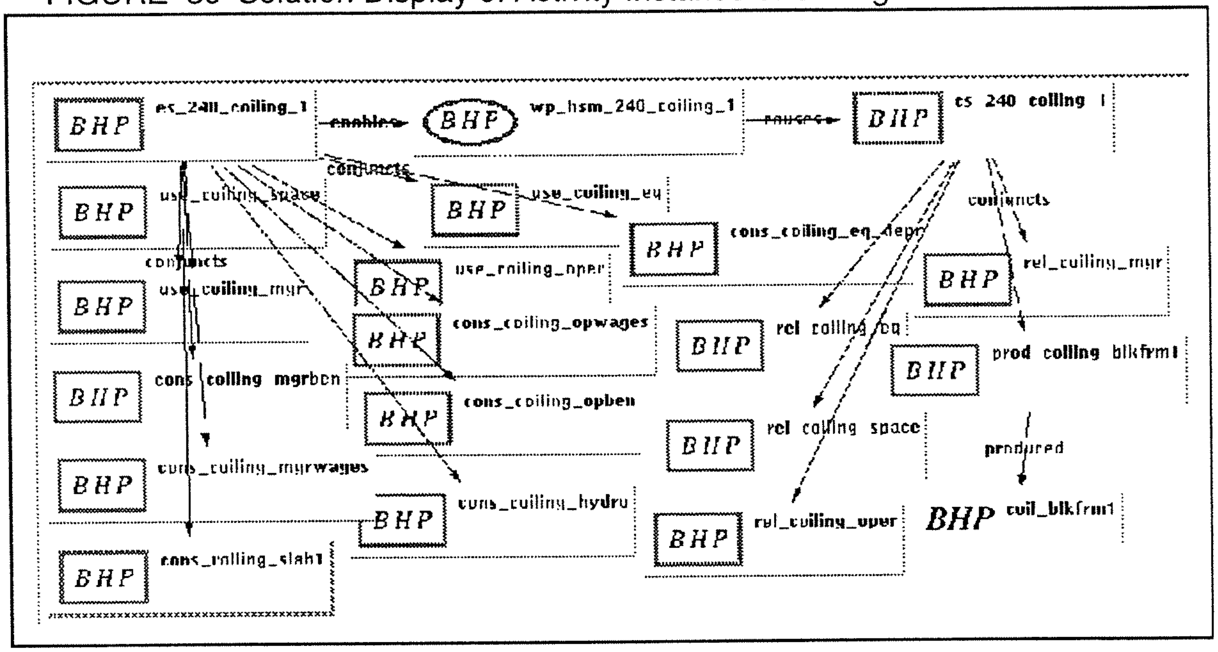


FIGURE 40 Solution re: Enabling States for Coiling Activity at #240

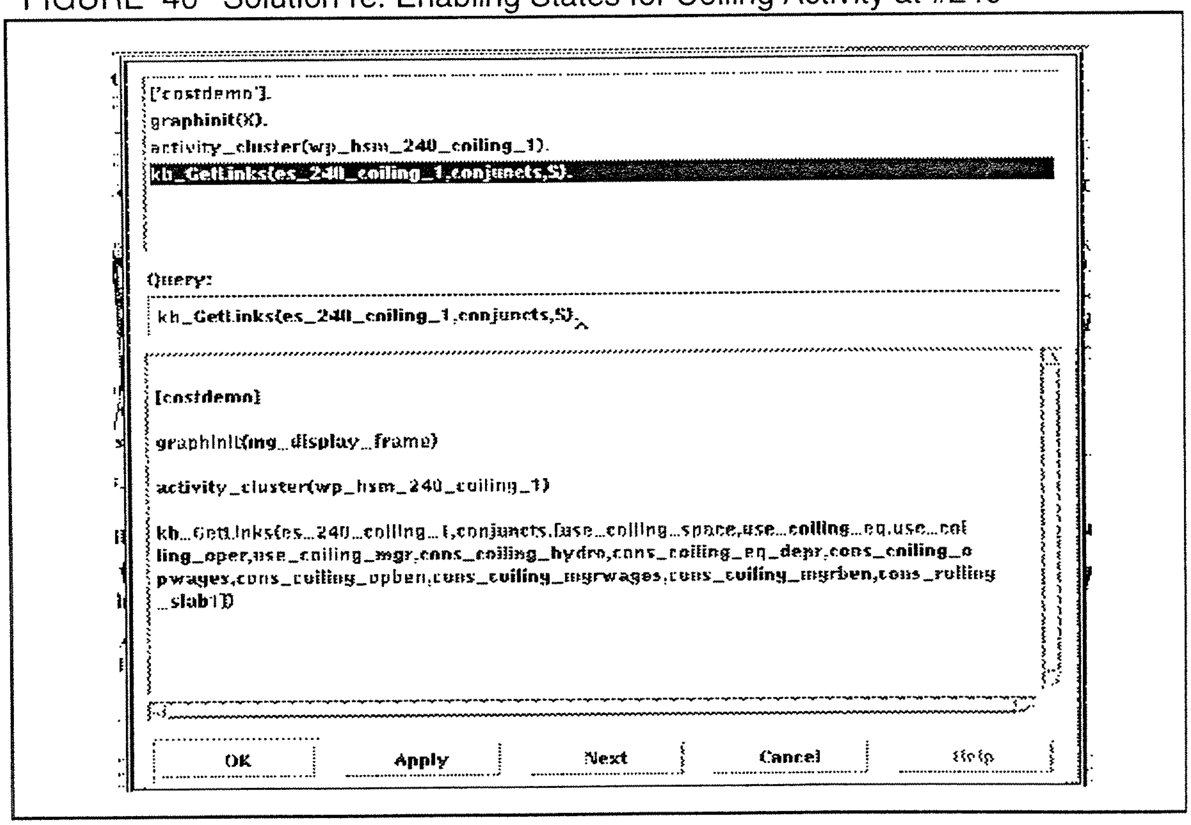


TABLE 24Resource Costs: Activity "wp_hsm_240_coiling"..... (Time =4 hrs)

Activity Name	*Resource*	*Resource Class (E = external, I = internal)*	*Resource consume/ usage specs. (qty q required)*	*Enabled Resource Cost Unit (v)*	*Enabled Resource cost of resource required*
wp_hsm_240_coiling	coiling_200th_space	I, non-period	1 unit	$ 240 / unit	* $ 320
	coiling_eq	E, non-period	1 unit	$ 0 / unit	$ 0
	coiling_oper	E, non-period	1 unit	$ 0 / unit	$ 0
	coiling_mgr	E, non-period	1 unit	$ 0 / unit	$ 0
	coiling_hydro	E, non-period	2,000 Kwh / hr	$ 0.25 / Kwh	$ 2, 000
[Internal Activity]	coiling_eq_depr	E, period	1 hr / hr	$ 50 / hr	$ 200
	coiling_opwages	E, period	1 hr / hr	$ 20 / hr	$ 80
	coiling_opben	E, period	1 hr / hr	$ 10 / hr	$ 40
	coiling_mgrwages	E, period	1 hr / hr	$ 40 / hr	$ 160
	coilingling_mgrben	E, period	1 hr / hr	$ 20 / hr	$ 80
	rolling_1_slab96	I, non-period	batch of 96 slabs	$32,490 for batch of 96	$ 32,490
				TOTAL COST	$ 35,370 for 96 units
	Cost Order produced.......... 96 units of coiled blackforms				

* Note: The provision of the 200,000 square feet space resource required for coiling is made possible through the activity, provide_coiling_200th_space. Given that the coiling activity takes 4 hours to complete 96 units of coiled blackform, the provision of space must be also made available for the 4 hours. For purposes of deducing the cost of factory space towards coiling, the event times of the consume states of the frontier resources indicated in Table 21 occur in the Prolog data base as:-

```
.........
occursT (enable (cons_coilspace_depr_recd, provide_coiling_200th_space), 17).
occursT (complete (cons_coilspace_depr_recd, provide_coiling_200th_space), 21).
occursT (enable (cons_coiltax_recd, provide_coiling_200th_space), 17).
occursT (complete (cons_coiltax_recd, provide_coiling_200th_space), 21).
........
```

Given the resource cost units of resources for the frontier activity, provide_coiling_200th_space, the cost of providing 200,000 square feet of factory space towards the coiling activity instance may be deduced through the cpa (cost point of activity) of the activity, provide_coiling_200th_space; or as the cpo (cost point of order) of the resource, coiling_200th_space, as follows:-

Prolog Query:- holdsT(project_act_costs(**provide_coiling_200th_space**, C), 22).

Solution: holdsT(project_act_costs (provide_coiling_200th_space, [0, 320, 0, 0, 320]), 22)

Prolog Query:- holdsT(project_order_costs (**coiling_200th_space**, C), 22).

Solution: holdsT(project_order_costs (coiling_200th_space, [0, 320, 0, 0, 320]), 22)

TABLE 25 Resource Costs: Activity "provide_coiling_200th_space" .. (Time = 4 hrs)

Activity Name	*Resource*	*Resource Class (E = external, I = internal)*	*Resource consume/ usage specs. (qty q required)*	*Enabled Resource Cost Unit (v)*	*Enabled Resource cost of resource required*
provide_coiling_200th_space	coilspace_depr_recd	E, period	1 hr / hr	$ 10 / hr	$ 40
	coiltax_recd	E, period	1 hr / hr	$ 20 / hr	$ 80
	coilinsur_recd	E, period	1 hr / hr	$ 30 / hr	$ 120
[Frontier Activity]	coilspace_heat	E, non-period	10 cu. m / hr	$ 1 / cu. m	$ 40
	coilspace_hydro	E, non-period	10 Kwh / hr	$ 1 / Kwh	$ 40
				TOTAL COST	$ 320 / unit
	Resource produced............1 unit of coiling_200th_space				

For both queries, the solutions are deduced through the Micro-theory of ABC - cpr, cpa and cpo give the solution as $320 for the supply of 200,000 square feet of factory space towards the coiling activity. These solutions are also equivalent to the sum of the cpr's (cost points of resources) summarized in Table 25.

[Note: The partial Prolog coding for cpr, cpa and cpo axioms for the implementation of the Micro-theory of ABC is shown in next Sec. 7.8]

For purposes of costing the coiling activity instance, wp_hsm_240_coiling_1, given the consume states of the resources of table 10, the cpa (cost point of activity) of wp_hsm_240_coiling_1 is queried and the deduced solution is as follows:-

Prolog Query:- holdsT(project_act_costs (wp_hsm_240_coiling_1, C), 22).

Solution: holdsT(project_act_costs (wp_hsm_240_coiling_1, [0, 35370, 0, 0, 35370]), 22)

Given that the 3,000 hours of scheduled production for 1997 would accommodate 750 activity instances for coiling (each activity instance being completed every 4 hours), then the total cost of all 750 instances for the activity class, wp_hsm_240 _coiling, is deduced from axiom 34 (chapter 4) as the aggregation of the cost point of each activity instance through the distinguishing predicate, cpa_subClass as follows:-

Prolog Query:- holdsT(cpa_class (wp_hsm_240_coiling, C), 3001).

Solution: holdsT(cpa_class (wp_hsm_240_coiling, 26527500), 3001)

The solution deduced indicates that the total cost of all instances towards the activity class, wp_hsm_240_coiling, is estimated to be $26,527,500 after 3,000 hours of sheduled 1997 coiled blackform production.

Solution: Estimated gross profit before taxes on 1997 sales revenue of $28 million for coiled blackform is approximately $1.5 million (= $28,000,000 - $26,527,500 = $1,472,500 ~ $1.5 million).

7.8 Partial Prolog Codes to implement cpr, cpa, cpo

```
..............................
............................
% We now include the axioms that aggregate the costs for each state of an
```

```
% activity given the costs for each state status in the effect axioms above.

holdsT(cpr(St,R,A,C), T) :-
holdsT(committed_res_cost(St,R,A,C1), T),
holdsT(enabled_res_cost(St,R,A,C2), T),
holdsT(disenabled_res_cost(St,R,A,C3), T),
holdsT(reenabled_res_cost(St,R,A,C4), T),
C is C1 + C2 + C3 + C4.

holdsT(dormant_act_cost(A,C), T) :-
activity_consumes(A,ConsumeResList),
committed_sum(A,ConsumeResList,C1,T),
activity_uses(A,UseResList),
committed_sum(A,UseResList,C2,T),
C is C1+C2.

committed_sum(A,[],0,T).
committed_sum(A,[Res|Tail], C,T) :-
committed_res_cost_unit(S,Res, A, Q),
holdsT(committed_res_cost(S,Res,A,C1), T),
committed_sum(A,Tail,C2,T),
C is C1+C2.

holdsT(execute_act_cost(A,C), T) :-
activity_consumes(A,ConsumeResList),
enabled_sum(A,ConsumeResList,C1,T),
activity_uses(A,UseResList),
enabled_sum(A,UseResList,C2,T),
C is C1+C2.

enabled_sum(A,[],0,T).
enabled_sum(A,[Res|Tail], C,T) :-
enabled_res_cost_unit(S,Res, A, Q),
holdsT(enabled_res_cost(S,Res,A,C1), T),
enabled_sum(A,Tail,C2,T),
C is C1+C2.

holdsT(suspend_act_cost(A,C), T) :-
activity_consumes(A,ConsumeResList),
disenabled_sum(A,ConsumeResList,C1,T),
activity_uses(A,UseResList),
```

```
disenabled_sum(A,UseResList,C2,T),
C is C1+C2.

disenabled_sum(A,[],0,T).
disenabled_sum(A,[Res|Tail], C,T) :-
disenabled_res_cost_unit(S,Res, A, Q),
holdsT(disenabled_res_cost(S,Res,A,C1), T),
disenabled_sum(A,Tail,C2,T),
C is C1+C2.

holdsT(reExecute_act_cost(A,C), T) :-
activity_consumes(A,ConsumeResList),
reenabled_sum(A,ConsumeResList,C1,T),
activity_uses(A,UseResList),
reenabled_sum(A,UseResList,C2,T),
C is C1+C2.

reenabled_sum(A,[],0,T).
reenabled_sum(A,[Res|Tail], C,T) :-
reenabled_res_cost_unit(S,Res, A, Q),
holdsT(reenabled_res_cost(S,Res,A,C1), T),
reenabled_sum(A,Tail,C2,T),
C is C1 + C2.

holdsT(cpa(A,C),T):-
holdsT(dormant_act_cost(A,C1),T),
holdsT(execute_act_cost(A,C2),T),
holdsT(suspend_act_cost(A,C3),T),
holdsT(reExecute_act_cost(A,C4),T),
C is C1+C2+C3+C4.

% Alternatively, we can compute cpa by aggregating the cpr for each
% resource utilized by the activity.

holdsT(check_cpa(A,C),T):-
activity_consumes(A,ConsumeResList),
cpr_sum(A,ConsumeResList,C1,T),
activity_uses(A,UseResList),
cpr_sum(A,UseResList,C2,T),
C is C1+C2.
```

```
cpr_sum(A,[],0,T).
cpr_sum(A,[Res|Tail],C,T):-
committed_res_cost_unit(S,Res,A,Q),
holdsT(cpr(S,Res,A,C1),T),
cpr_sum(A,Tail,C2,T),
cpr_sum(A,Tail,C2,T),
C is C1+C2.

% We now aggregate the costs of activities that are executed for an order.

holdsT(cpo(Order,C),T):-
order_executes(Order,ActivityList),
activity_sum(ActivityList,C,T).

activity_sum([],0,T).
activity_sum([A|Tail],C,T):-
holdsT(cpa(A,C1),T),
activity_sum(Tail,C2,T),
C is C1+C2.

% We now aggregate the costs for all instances in a class of activities or
% class of orders.

holdsT(cpa_class(Class,C),T):-
instance_list(Class,ActivityList),
activity_sum(ActivityList,C,T).

holdsT(cpo_class(Class,C),T):-
instance_list(Class,OrderList),
order_sum(OrderList,C,T).

order_sum([],0,T).
orer_sum([A|Tail],C,T):-
holdsT(cpo(A,C1),T),
order_sum(Tail,C2,T),
C is C1+C2.
```

7.9 Concluding Remarks

This chapter has highlighted the benefits derived from the Principle of Resource Probing. With the "commit envelope" it provides the trace of processes leading to the supply of an internal resource. The Micro-theory provides the cost view of these processes. This information of "commit envelope" along with the associated costs of processes, should prove invaluable towards the "make-buy" decisions of resources. The deduced cost of the resource supplied in-house is made possible through the Micro-theory of ABC, the Micro-theory of Resource Cost Units and the Formalization of Overheads, and *includes all direct, indirect and overhead costs borne by the company if it were to continue producing the resource in-house.* Noting that major corporations are presently going towards "out-sourcing", one may consider the Principle of Resource Probing, the Micro-theory of ABC, the Micro-theory of Resource Cost Units and the Formalization of Overheads as being major contributions that are not only timely, but extremely practical and useful to corporations. From the point of implementation, all costs are traced directly to "point of absorption" without messing and guessing about "drivers", as is the case with ABC developments to date.

Finally, the BHP study gives credence to the Cost Advisor being able to provide invaluable assistance towards the solving of fairly complex strategic cost management questions that may be posed by a corporation.

CHAPTER 8 Re-usability

8.1 Introduction

Our first measure of success of a representation has been competence: how well does the representation support problem solving? That is, what questions (aka competency questions) can the representation answer or what tasks can it support?

The second measure of success of a representation is the extent to which the existing representation (aka native ontology) is re-usable. One may view *re-usability* from two perspectives: 1) the *applications' perspective*, and 2) the ontological perspective based on the *reducibility concept.*

The *applications' perspective* of re-usability is based on the extent to which the native ontology can successfully perform tasks in the knowledge domain for two or more different enterprises.

The *reducibility concept* is the extent to which other representations (aka target ontologies) of the knowledge domain can be expressed in terms of the native ontology to do some of the tasks performed by the target ontologies. Assume that a target ontology can perform some tasks based upon some "intended interpretations" of the target ontology. Then, based on the reducibility concept, if the target ontology can be reduced to the native ontology that can do those tasks based upon "equivalent intended interpretations", and keeping in mind that the native ontology design was unaware of those tasks of the target ontology, then this would prove that the native ontology can be "re-used" for the "unknown tasks" of the target ontology. Hence, this research considers the application of the reducibility concept as a means to prove the re-usability of the Cost Ontology, i.e. the native ontology.

Suppose we are given different enterprises that use different target representations to perform tasks. If each of the target representations with "intended interpretations" are reducible to a native representation with "equivalent intended interpretations" that can perform the required tasks for each of the enterprises considered, we would be providing evidence for the usage of the native ontology in different enterprises, thereby promoting or complementing the aspect of re-usability from an applications' perspective.

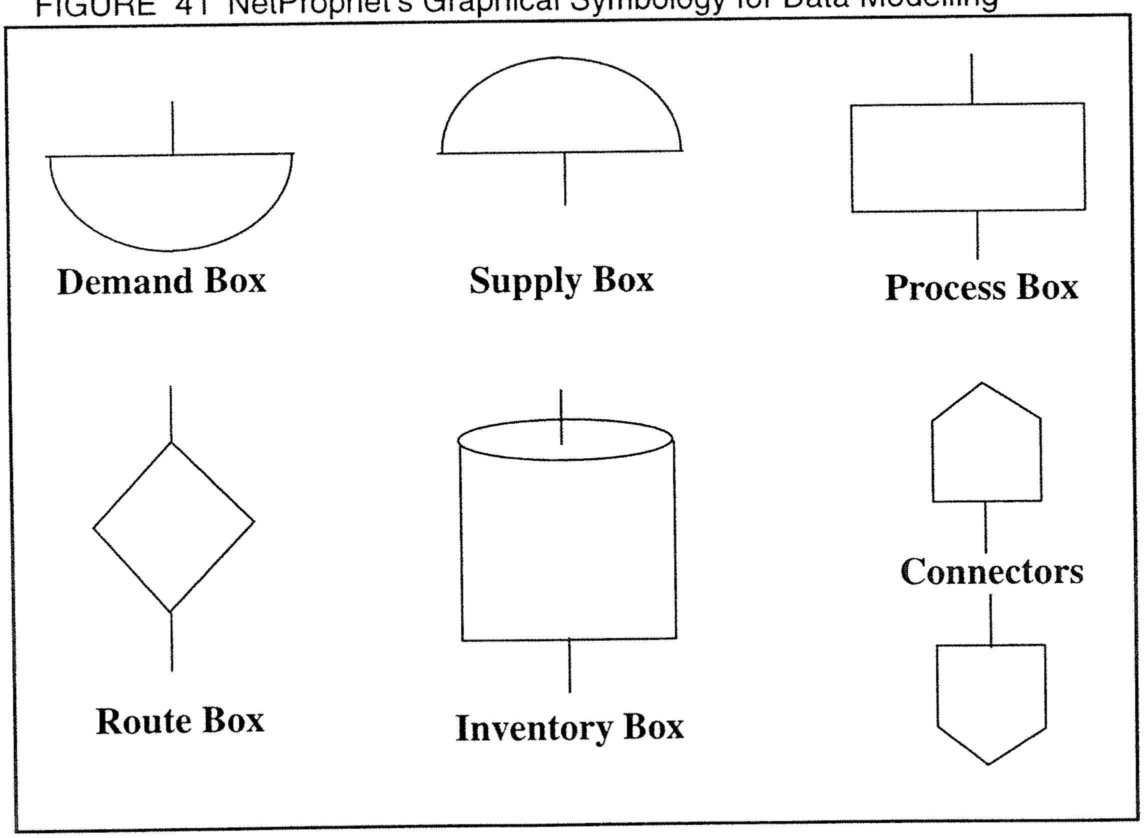
The approach to reducibility [Gruninger 96] taken in this report is the one proposed by Michael Gruninger of EIL. The proposed approach is premised upon ontologies being able to solve competency questions. Suppose that we are given some set of competency questions which are specified using the representation of the target ontology. If these competency questions can be reduced to solvable competency questions with some equivalent intended interpretations based on the native ontology, then we would have demonstrated the reducibility aspect of the native ontology.

The extent of re-usability of the Cost Ontology for TOVE from an applications' perspective has been demonstrated by successfully modelling the deHavilland and the BHP enterprises for a forensic cost analysis with ABC at deHavilland (Chapter 6), and the solving of cost related queries for BHP (Chapter 7). The aspect of reducibility is addressed in this chapter where competency questions that use NetProphet software (reviewed in Ch. 2) representations in the domain of activity-based management are translated or reduced to solvable competency questions based on the Cost Ontology for TOVE.

8.2 NetProphet Terminology and Representation

NetProphet's design approach [Albright 97] for developing an ABC model resembles industrial engineering applications in which the following takes place - inputs are linked to processes; process choices are made; constraints are considered; inventory levels are maintained; and outputs are defined. Model construction involves top-down approach, beginning with resources and terminating with the demand function. Netprophet uses a graphical modelling schema to "clarify interrelationships between the demands, activities, and resources of the project area" **[Netprophet Applications Guide, 96]**. The Netprophet representation model is constructed of five different graphical symbols (boxes) to represent the various components within a project area. The five types of boxes used are the Supply, Process, Route, Inventory, and Demand boxes, and Connectors are used to indicate links between the boxes (Figure 41).

FIGURE 41 NetProphet's Graphical Symbology for Data Modelling



The *concepts and meanings* **[Netprophet Applications Guide 96, Albright 97]** of Demand Box, Process Box, Supply, Route, Inventory, and entities are explained in the natural language. Later in this chapter, these explanations are formalized in FOL to establish a "NetProphet Ontology" (aka target ontology) with a view to reducing the target ontology to the TOVE Ontology (aka native ontology).

1. Demand Box: represents the outputs of a process_box (e. g., a manufactured product, an internal or external service. Demand boxes are used to represent the final cost objects of the model.
2. Process Box: it can represent a resource, an activity or both.
3. Supply Box: represents supply of a resource (e.g., raw materials, a machine, assembly operator, a nurse, etc.) required by a process box.
4. Route Box: when there is a choice of required inputs to a process box, e.g., a capstan lathe machine or a turret lathe machine is required for a machining activity, the Route Box is used.
5. Inventory Box: It is used to carry items forward from one time period to the next (e.g., finished goods assembled in a period may not be sold in that period, and must be carried forward as inventory). An inventory box has one entry link - from a process box or a supply box depending upon whether the inventoried resource is produced by a process box or supplied by a supply box. A quantity value and an attribute (e.g., supply_box_id, process_box_id) are typically associated with an inventory box.
6. Connectors: although *not a modelling symbol*, a connector is used to show where links or a relationship exists between the boxes in the schematic.

8.3 Reducing NetProphet's Representation

It this section, a data model for a hospital scenario based on NetProphet's representation is presented. This is followed by an explanation of the data model in the context of the hospital scenario. From an ontological engineering perspective, this research measures the competency of an ontology based upon its ability to provide solutions to questions (aka competency questions). These competency questions can be looked upon as "tasks" that must be satisfied by the ontology. Therefore, based on NetProphet's representation, some ABC competency tasks are then put forth for Netprophet in the context of the hospital scenario. Next, a reduction of the ABC tasks for the Netprophet representation to ABC tasks for the Cost Ontology in TOVE is attempted by:- (i) formalizing a Netprophet Ontology in FOL; (ii) defining axioms in FOL that "translate" or show the equivalence between the Netprophet Ontology and that of TOVE; (iii) reducing Netprophet's hospital scenario data model to activity clusters in TOVE; and (iv) reduce the ABC tasks for Netprophet in terms of activities, resources, states, cost order, cost point of resource (cpr), cost point of activity (cpa) and cost point of order (cpo) for the Cost Ontology in TOVE.

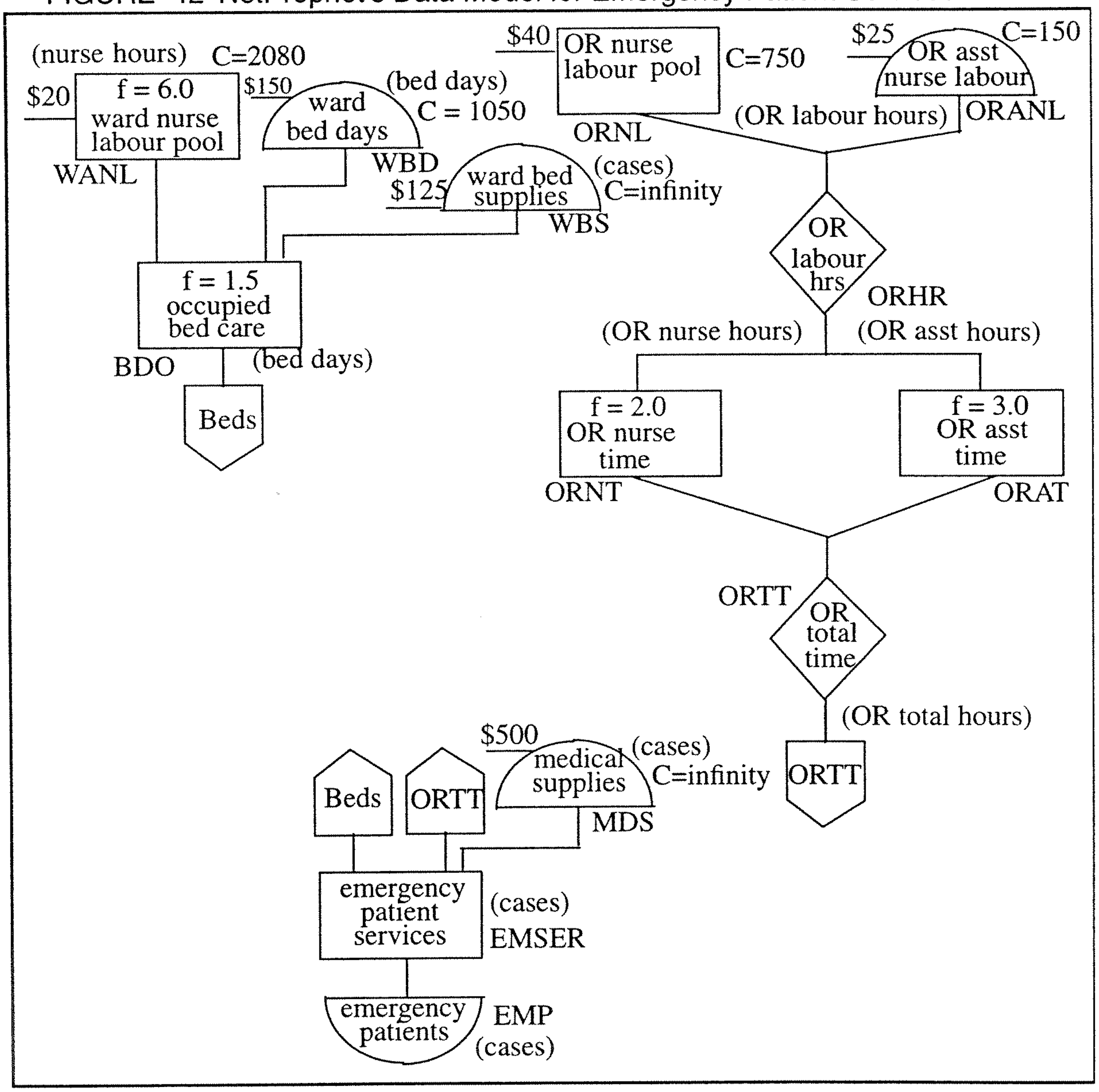
8.3.1 NetProphet Data Model: Hospital Scenario

The following figure is an example of a data model for a hospital scenario based on NetProphet's representation. The modelling diagram of the data model for the hospital represents interrelationships amongst the resources, activities and demands based on the usage of the demand box, supply box, process box, route box and connectors.

The hospital scenario:

Based on the data model, the hospital scenario may be interpreted as follows: A hospital caters to the demands of emergency patient cases (demand box - EMP) admitted by providing emergency patient services (process box - EMSER). These services require resources such as medical supplies (supply box - MDS), operating room nurses from the OR nurse pool (process box - ORNL) or the OR assistant nurse labour supply (supply box - ORANL), ward beds (supply box - WBD), ward bed supplies (supply box - WBS), and ward nurses from the ward nurse labour pool (process box - WANL) that process ward beds occupied (process box - BDO) by the emergency patients. The requirements for bed occupancy and OR total time towards emergency patient services from the supply side of the model are linked through connectors Beds and ORTT respectively to the corresponding connectors from the demand side of the model represented by demand box - EMP.

FIGURE 42 NetProphet's Data Model for Emergency Patient Services



The available capacity, time and cost data for emergency patient cases admitted are as follows:

- cost of medical supplies is $500 per case (supply box - MDS, capacity C = infinity)
- average ward bed occupancy is 1.5 bed days per case (process box - BDO, f = 1.5)
- each occupied bed day requires 6.0 ward nurse hours @ $20 per ward nurse hour (process box - WANL, f = 6.0, available capacity C = 2,080 ward nurse hours, $20)
- the supply capacity of ward beds is 1,050 bed days @ $150 per bed day (supply box - WBD, C=1050, $150)
- each case requires ward bed supplies costing an average of $125 per case (supply box - WBS, C = infinity)
- based on averages, each case requires: 2 hours OR nurse time (process box - ORNT) drawn from regular OR nurse labour pool of capacity 750 hours @ $40 per hour (process box - ORNL, C = 750, $40), or 3 hours OR assistant nurse time (process box - ORAT) drawn from a supply of OR assistant nurse labour with capacity of 150 hours @ $25 per hour (supply box - ORANL, C = 150, $25)

8.3.2 ABC Competency Tasks for NetProphet

Based on NetProphet's representation, some ABC competency tasks for NetProphet for the hospital scenario are:

1. Stating the query in lay-person's terminology: What is the cost of occupied bed care per patient case?

   What are the costs from process box WANL and supply box WBD towards unit demand of demand box EMP?

2. Stating the query in lay-person's terminology: What is the cost of OR care services per patient case?

What are the costs from process box ORNL and supply box ORANL towards unit demand of demand box EMP?

3. Stating the query in lay-person's terminology: What is the cost of emergency patient services per patient case?

   What is the cost per unit demand of demand box EMP?

8.3.3 Formalizing Target Ontology re: Netprophet "Ontology"

Within the literature for Netprophet, there is no explicit acknowledgment of an ontology as has been defined within this research. However, *the Supply, Process, Route, Inventory, and Demand boxes being the representational entities within the Netprophet model, we consider them as forming a core Netprophet Ontology (i.e. a target ontology) from the ontological perspective of this research.*

NetProphet does not have a defined ontology as we understand it. At best, NetProphet has a terminology that lacks preciseness and brings forth some ambiguity between resources and activities represented by the process box.

In order to formalize a Netprophet Ontology, the *intuitive concepts and meanings* **[Albright 97]** of Supply, Process, Route, Inventory, and Demand entities are explained in the natural language, and then expressed as FOL axioms denoted as Net_Axioms.

It should be noted that Netprophet does not state any specific relationships or predicates. For these "missing pieces" and based upon the "intended interpretations" gathered based upon inter-actions with the Sapling Corporation and a study of [Sapling 97], use of some Cost Ontology predicates are use to fill in for the "missing pieces".

1. Demand (Box): represents the outputs of a process_(box) (e. g., a manufactured product, an internal or external service. In all situations, s, Demand Boxes are used to represent the final cost objects of the model.

Net_Axiom: **$\forall$ *(d, s) holds(demand_box(d), s)* $\supset \exists$ *p, holds(process_box(p), s)* $\wedge$**
_holds (has_cost_object(p, d), s)_ **(FOL 113)**

2. Supply (Box): represents supply of a resource (e.g., raw materials, a machine, assembly operator, a nurse, etc.) required by a process_(box) in all situations s.

Net_Axiom: $\forall$ ***(r, s) supply_box(r)*** $\equiv$ ***resource(r)*** $\supset \exists$ ***p, process_box(p)*** $\wedge$ ***holds (requires(p, r), s)*** (FOL 114)

3. Process (Box): it can represent a resource or an activity in situations s.

Net_Axiom: $\forall$ ***s, p, holds(process_box (p), s)*** $\equiv$ ($\exists$ ***r, a), holds (resource (r), s)*** $\vee$ ***holds(activity (a), s]*** (FOL 115)

4. Route (Box): represents a situation *s* when there is a choice of required inputs to a process_box (e.g., a capstan lathe machine or a turret lathe machine is required for a machine_process).

Net_Reduce_Axiom (Schema): $\forall$***b, p, s, holds(route_box (b, p), s)*** $\supset$ ($\exists$ ***r1, r2, r3, ... rn), holds (process_box (p), s)*** $\wedge$ ***[holds(requires(p, r1), s)*** $\vee$ ***holds(requires(p, r2), s)*** $\vee$ ***holds(requires(p, r3), s)*** $\vee$ ***....*** $\vee$ ***holds(requires(p, rn), s)]*** (FOL 116)

5. Inventory (Box): It is used to carry items forward from one time period to the next (e.g., finished goods assembled in a period may not be sold in that period, and must be carried forward as inventory). An inventory_box has one entry link:- from a process_box or a supply_box depending upon whether the inventoried resource is produced by a process_box or supplied by a supply_box. A quantity value and an attribute (e.g., supply_box_item_class, process_box_item_class) are associated with an inventory_box. From the perspective of this research, we define inventory_box as a *fluent denoted as inventory_box (r, q)* that holds true in a situation s, where r denotes a class of inventoried resource item produced by a process_box or supplied by a supply_box, and q denotes the quantity in inventory that changes from situation to situation.

Net_Axiom: $\forall$ ***r, q, s, holds(inventory_box (r, q), s)*** (FOL 117)

8.3.4 Reduction Axioms: NetProphet Ontology --> TOVE Ontology

In this section, axioms (indicated as Net_Reduce_Axiom) that enable us to "translate" or reduce the NetProphet Ontology (target ontology) with intended interpretations of the previous section to the TOVE Ontology with equivalent intended interpretations, are put forth.

1. Process Box: **process_box (p)**

   The object class process_box (p) of the target ontology is equivalent to the object class of resource(r) or activity(a) in TOVE in all situations s.

Net_Reduce_Axiom: $\forall$ ***s, p, holds(process_box (p), s)*** $\equiv$ ($\exists$ ***r, a), holds (resource (r), s)*** $\vee$ ***holds(activity (a), s]*** **(FOL 118)**

2. Demand Box: **demand_box (d)**

   In TOVE, that which is produced by the caused state of an activity is the cost order or the reason for performing that activity. Further, in TOVE, the cost order of an activity may be a resource to some other activity. The object class - demand_box(d) in the target ontology is equivalent to the class of objects produced in the caused state of an activity.

Net_Reduce_Axiom: $\forall$***d, s, holds(demand_box(d), s)*** $\supset$ $\exists$ ***p, holds (process_box(p), s)*** $\wedge$ ***holds(has_cost_object(p, d), s)*** $\equiv$ $\exists$ ***a, r, holds(activity (a), s)*** $\wedge$ ***holds(resource (r), s)*** $\wedge$ ***holds(produces(a, r), s)*** $\supset$ ***holds(has_cost_order (a, r), s)*** **(FOL 119)**

3. Supply Box: **supply_box (r)**

   The object class supply_box (r) of the target ontology is equivalent to the object class resource (r) in TOVE.

   Net_Reduce_Axiom: $\forall$***r, supply_box (r)*** $\equiv$ ***resource (r)*** **(FOL 120)**

4. Route Box: **route_box (b, p):**

   In TOVE, a situation s when an activity disjunctively requires one of many alternative resources is equivalent to the situation when a route_box of the target ontology holds true.

Net_Reduce_Axiom (Schema): $\forall$ *(b, p, s, r1, r2, r3, ... rn, a), holds(route_box (b, p), s)* $\equiv$ *holds (activity (a), s)* $\wedge$ *[holds(requires(a, r1), s)* $\vee$ *holds(requires(a, r2), s)* $\vee$ *holds(requires(a, r3), s)* $\vee$ *....* $\vee$ *holds(requires(a, rn), s)]* (FOL 121)

5. Inventory (Box): **inventory_box (r, q)**

In order to establish an equivalence between the *fluent inventory_box(r, q)* of the Netprophet ontology to that of the TOVE ontology, we use the class of resource based producer/consumer actions defined in TOVE with respect to the *resource point fluent denoted as rp(r, q)* [Fadel et al. 94]. The classes of TOVE "states" are really relationships among resources and activities. The enabling states of an activity consume or use a quantity of resource, thereby decreasing the level of inventory. On the other hand, the caused state of an activity produces a quantity of resource, thereby increasing the level of inventory. Hence, the level or quantity of an inventoried resource represented as a resource point fluent, denoted *rp(r, q)*, will change from situation to situation. Therefore, intuitively there is an "ontological connectivity" between the concept of the inventory_box in Netprophet and the concept of the resource point in TOVE.

Formalized reduction axioms in FOL are stated as follows and make use of the resource point fluent as used to define Resource-based Effect axioms for the classes of TOVE activities [Gruninger 95] depending upon whether the resource is consumed, used, or produced by the activity.

Net_Reduce_Axiom: $\forall$ *r, q, s, holds(inventory_box (r, q), s)* $\supset \exists$ *a, q', q", s', [consumes(a, r)* $\vee$ *uses (a, r)* $\vee$ *produces (a, r)]* $\wedge$ *quantity(a, r, q')* $\equiv$ *[holds(rp(r, q"), s']* $\wedge$ *Do(a, s', s)* $\supset$ *holds(rp(r, q" - q'), s]* $\wedge$ *(q = q" - q')* (FOL 122)

8.3.5 Comments re: NetProphet & TOVE Representations

The core ontology in TOVE draws a strong delineation amongst the objects - activity, state, and resource representation - through precise definitions and distinct graphical symbologies. Unlike TOVE, NetProphet's process box may represent activities and resources.

NetProphet's concept of the demand box is akin to the concept of the cost order in the Cost Ontology for TOVE. The demand box in NetProphet may be viewed as the reason to draw upon the resources represented by supply boxes and process boxes. The route box in NetProphet has some semblance to the more precise representation of conjunct and disjunct states in TOVE. While connectors in NetProphet's data model link objects such as process box, supply box and route box, the TOVE ontology uses precise relationships and their inverse relationships such as: enables and enabled_by, causes and caused_by, has_subactivity and subactivity_of, consumes and consumed_by, conjuncts and conjunct_of, disjunct and disjunct_of, consumes and consumed_by, produces and produced_by, etc., to link the TOVE objects - activity, state, resource - of the activity clusters in the enterprise model.

8.3.6 Reducing a NetProphet Task to TOVE's Cost Ontology Task

TOVE's Activity Clusters for NetProphet's data model of the Hospital Scenario are shown in Figure 43. The ems_aggregation activity has sub-activities: ward_bed_care and or_care.

The ward_bed_care activity has cost order ems_ward_care_patient and requires the resources: ward_bed_days (enabled resource cost unit = $150 per bed day, discrete quantity consumption = 1.5 bed days per case); ward_nurse_hrs (enabled resource cost unit = $20 per nurse hour, discrete quantity consumption = 6 nurse hours per case); and ward_bed_supplies (enabled resource cost unit = $125 per unit bed supply, discrete quantity consumption = 1 unit bed supply per case). For this example, note that resource, ems_ward_patient, is considered an external resource with no cost contribution to the ward_bed_care activity.

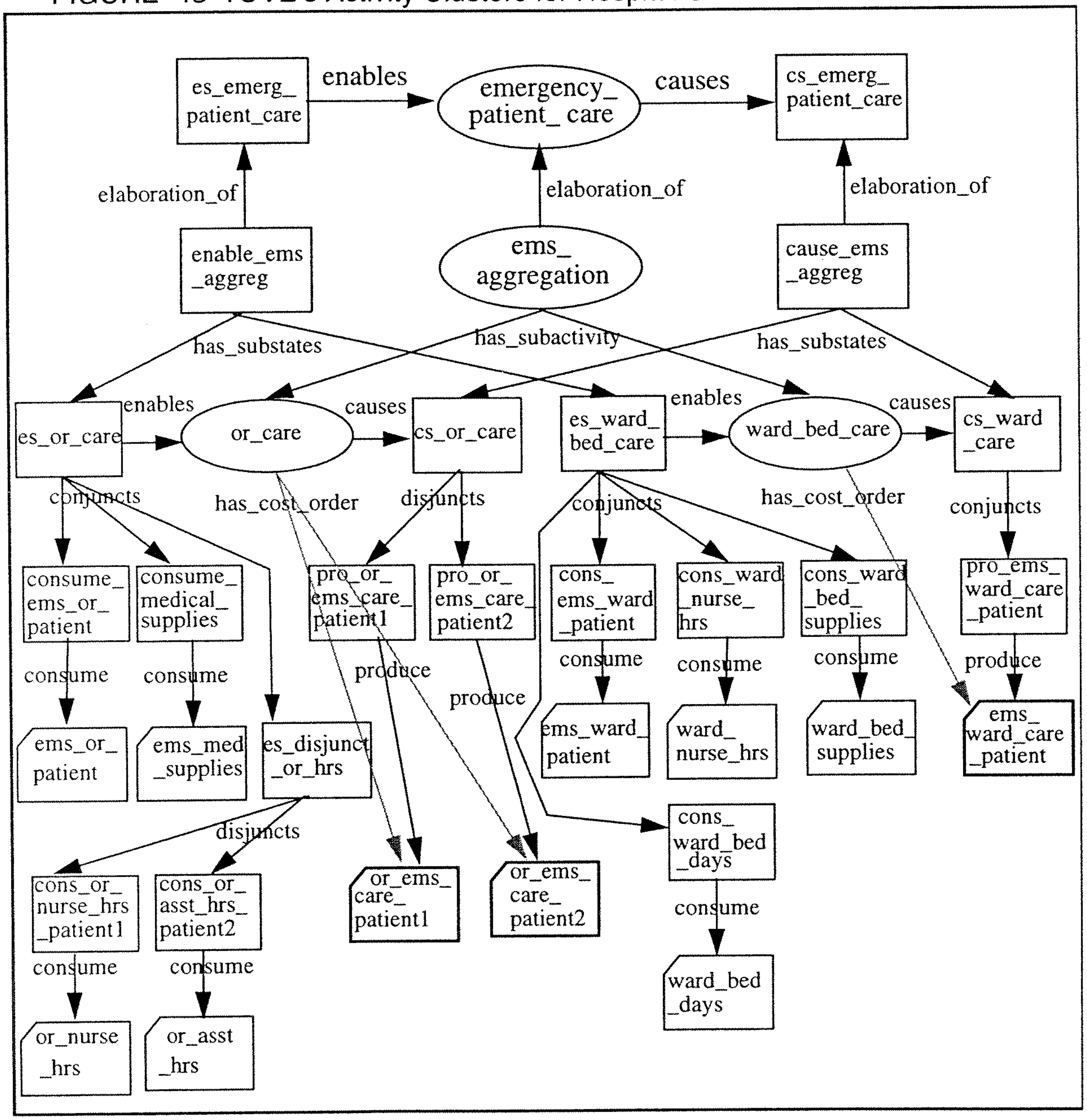
As an example, NetProphet's competency task question:-

> *What are the costs from process box WANL and process box BDO towards unit demand of demand box EMP?*

maybe posed more precisely in the Cost Ontology for TOVE as:

- **What is the cost point of cost order (cpo) ems_ward_care_patient at time t?**

FIGURE 43 TOVE's Activity Clusters for Hospital Scenario



8.3.7 Solving Reduced NetProphet's Task with Micro-theory of ABC

In Chap. 4, based on Axiom schema (FOL 46), the cost point of a cost order is equivalent to the cost point of activities (cpa) that has_cost_order relationship to the cost order. Hence, based on the activity cluster model in Figure 43, the solution to the cpo(c, x, t) task, where the cost order x = ems_ward_care_patient is equivalent to the cpa of activity, ward_bed_care at time t.

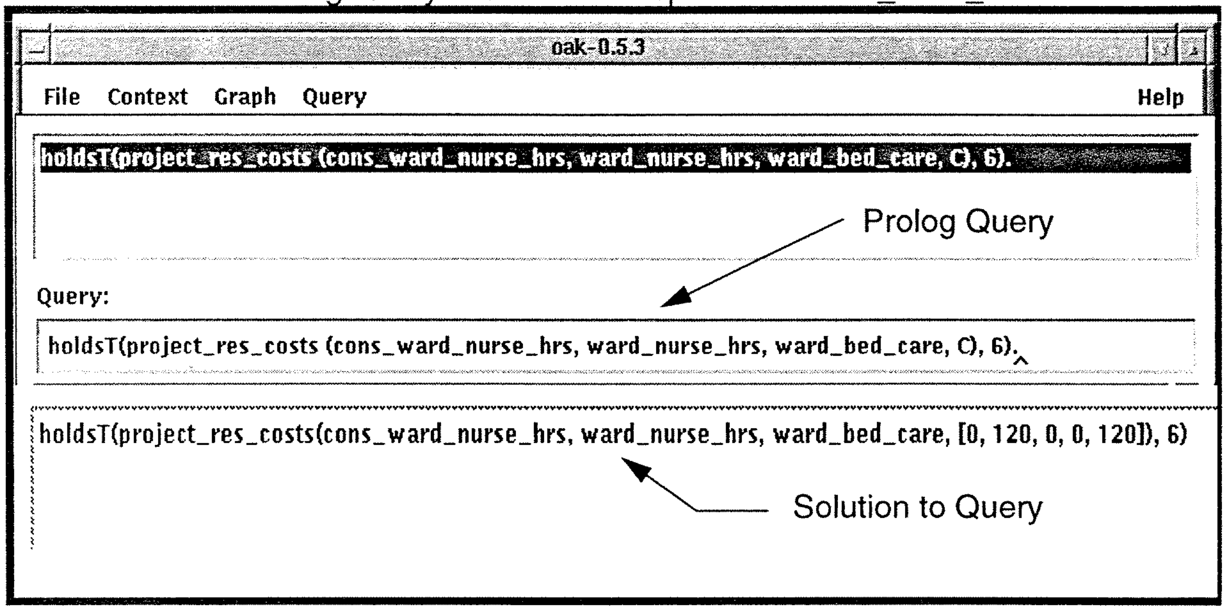
Also, in Chap. 4, based on Axiom schema (FOL 4), the cpa of an activity at time t is equivalent to the sum of cost point of resources (cpr) required by the activity at time t. The activity cluster model of Figure 43 indicates the solution to the cpa(a, c, t) task, where activity a = ward_bed_care is equivalent to the sum of cpr's of resources: ward_bed_days, ward_nurse_hrs and ward_bed_supplies. Since resource cost units for these resources are known, the cpr, cpa and cpo may be deduced relevant to the cost order ems_ward_care_patient.

A portion of the Prolog knowledge base for the implementation input of the ward_bed_care activity would be as follows assuming the activity occurs at time point 1 and is completed at time point 5:-

```
:- multifile holdsT/2, occursT/2, concurrent/2.
% activity_occursT(ward_bed_care, 1).
occursT (enable (cons_ems_ward_patient, ward_bed_care), 1).
occursT (enable (cons_ward_nurse_hrs, ward_bed_care), 1).
occursT (enable (cons_ward_bed_days, ward_bed_care), 1).
occursT (enable (cons_ward_bed_supplies, ward_bed_care), 1).
occursT (complete (cons_ems_ward_patient, ward_bed_care), 5).
occursT (complete (cons_ward_nurse_hrs, ward_bed_care), 5).
occursT (complete (cons_ward_bed_days, ward_bed_care), 5).
occursT (complete (cons_ward_bed_supplies, ward_bed_care), 5).
..............
.............
holdsT(enabled_res_cost_unit (ems_ward_patient, 0), 5).
holdsT(enabled_res_cost_unit (ward_nurse_hrs, 20), 5).
holdsT(enabled_res_cost_unit (ward_bed_supplies, 125), 5).
holds((enabled_res_cost_unit (ward_bed_days, 150), 5).
concurrent (a, b).
```

For example, the cpr for resource ward_nurse_hrs is obtained from the query and solution as shown in the computer output shown in Figure 44.

FIGURE 44 Prolog Query & Solution re: cpr for resource_ward_nurse



The solution of Figure 44 indicates that the cpr for resource ward_nurse_hrs is $120 at time 6.

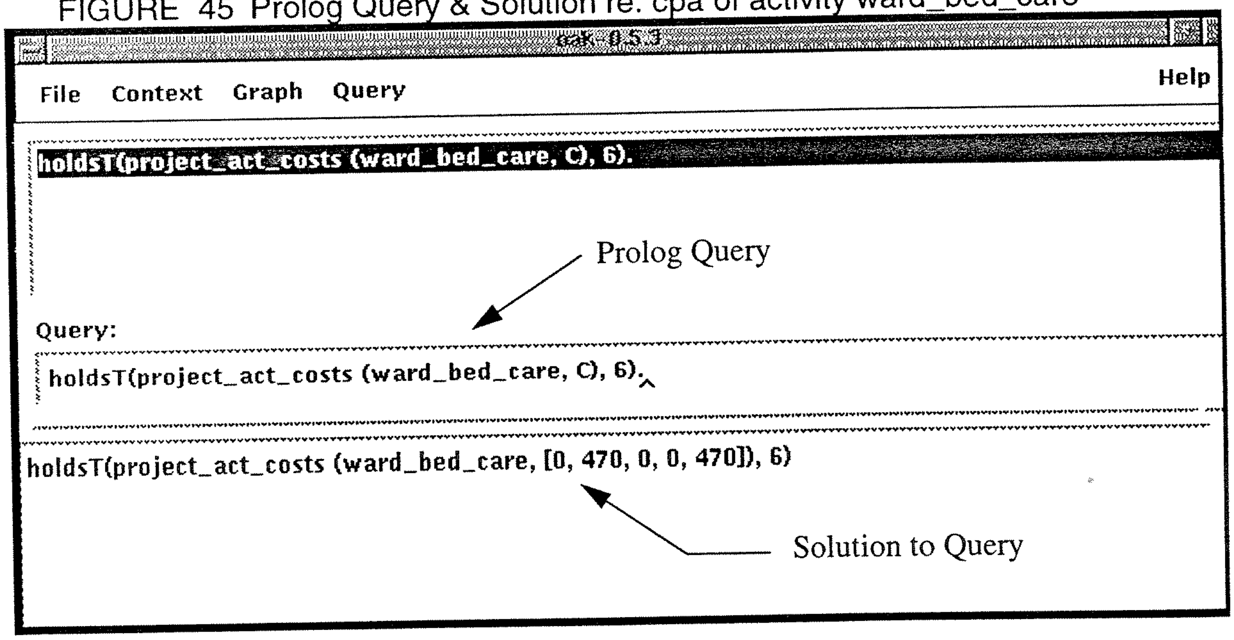
The following table is a summary of cpr's for activity ward_bed_care:-

TABLE 26 Deduced cpr's for Activity ward_bed_care

Activity Name	*Resource*	*resource discrete consume specs*	*enabled_res _cost_unit ($/unit)*	*cpr at t = 6 ($)*
ward_bed_care	ems_ward_patient	1 unit	0	0
	ward_nurse_hrs	6 units	20	120
	ward_bed_supplies	1 unit	125	125
	ward_bed_days	1.5 units	150	225

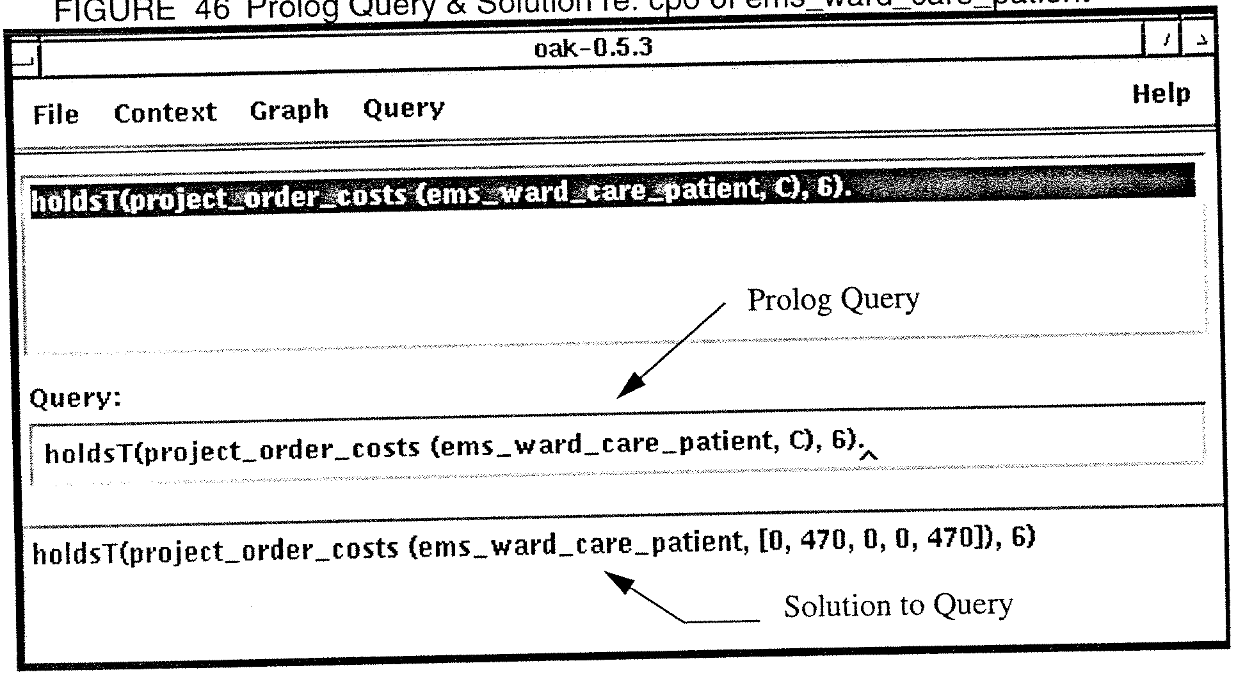
The cpa for activity ward_bed_care is obtained from the query and solution as shown in the computer output shown in Figure 45.

FIGURE 45 Prolog Query & Solution re: cpa of activity ward_bed_care



The solution of Figure 45 indicates that the cpa of ward_bed_care at t = 6 when 1 ems_ward_patient case is completed is $470.

FIGURE 46 Prolog Query & Solution re: cpo of ems_ward_care_patient



The cpo of cost order ems_ward_care_patient is obtained from the query and solution as shown in the computer output of Figure 46. The solution indicates that the cpo of ems_ward_care_patient at t = 6 is produced at a cost of **$470 per patient.**

Note that the Netprophet solution to this query also checks out to be $470 as follows [refer to Figure 42]:-

- for each demand (i.e., each patient) from demand box EMP, 6 hours of ward nurse labour pool is used @ $20 per hour of ward nurse labour; therefore, the cost contribution to each patient due to ward nurse labour input is *$120 / patient =* [($20/ hour)(6 hours / patient)] ..... cost contribution from process box WANL;

- for each demand (i.e., each patient) from demand box EMP, BDO process box calls for 1.5 bed days of occupied bed care @ $150 per ward bed day, the bed days being drawn from supply box WBD; therefore, the cost contribution to each patient due to of ward bed days required is *$225*/patient = [($150 /bed day) (1.5 bed days)] .... partial cost contribution from process box BDO;

- for each demand (i.e., each patient) from demand box EMP, ward bed supplies from supply box WBS are consumed @$125 per patient or per case; therefore, the cost contribution to each patient due to ward bed supplies consumed is *$125* .... partial cost contribution from process box BDO
- cost contribution from supply box WANL .... = $120 per patient

total cost contribution from supply box BDO = $225 + $125 = <u>$350 per patient</u>

Total cost per patient = **<u>$470 per patient</u>**

Similarly, the other NetProphet competency task queries maybe reduced to tasks in the Cost Ontology for TOVE that are solvable with the Micro-theory for ABC in TOVE.

8.4 Concluding Remarks

This research has viewed *re-usability* from two perspectives: 1) the *applications' perspective*, and 2) the ontological perspective based on the *reducibility concept.*

The *applications' perspective* of re-usability is based on the extent to which the Cost Ontology in TOVE successfully performs tasks in the knowledge domain for two or more different enterprises. This has been demonstrated with the deHavilland and BHP case studies.

The *reducibility concept* is the extent to which other representations (aka target ontologies) of the knowledge domain can be expressed in terms of the Cost Ontology (aka native ontology) to do some of the tasks performed by the target ontologies.

The reducibility concept has been applied to the representations used by the activity based management software NetProphet [Sapling 96]. NetProphet does not have a precise ontology as is presented in this report, but does have terminologies and modelling representations ("target ontologies") to perform its competency tasks.

Reducibility has been demonstrated with regard to NetProphet by:- (i) formalizing a Netprophet Ontology in FOL; (ii) defining axioms in FOL that "translate" or show the equivalence between the Netprophet Ontology and that of TOVE; (iii) reducing Netprophet's hospital scenario data model to activity clusters in TOVE; and (iv) reducing the ABC tasks for Netprophet in terms of activities, resources, states, cost order, cost point of resource (cpr), cost point of activity (cpa) and cost point of order (cpo) for the Cost Ontology in TOVE.

Having applied the reducibility concept to NetProphet, this chapter may be viewed as having developed a procedure for applying the reducibility concept between target and native ontologies in practice. In summary, the procedure involves the following steps:-

1. Formalize the target ontology in a precise and unambiguous language such as FOL,
2. Define axioms that "translate" or show equivalence between a native (in our case the TOVE Ontology) and a target ontology (e.g., NetProphet Ontology),
3. Reduce a data model representation for the target ontology to one for the native ontology,
4. Reduce tasks in a domain (e.g., the cost management domain) applicable to the target ontology to tasks in the same domain for the native ontology.

This chapter puts forth a strong case in justifying the re-usability of the Cost Ontology in TOVE for the domain of enterprise modelling and activity-based cost management.

CHAPTER 9 Conclusion, Contributions, Future Research

9.1 Conclusion

The thesis of this dissertation is that costs within an organization can be described by representing them in an enterprise model to provide a Formal Model and a Micro-theory of ABC; and costs of a resource can be objectively deduced to include direct, indirect, and overhead costs for a populated enterprise model by applying a Micro-theory of ABC and a Theory of Resource Cost Units.

The research has been motivated by the two fundamental problems in the ABC Principle. The widely accepted ABC Principle includes the assignment of cost to activities based on their use of resources, and the assignment of costs to "cost objects"[1] based on their use of activities [Brimson 91]. Essential for this costing principle to be applied, one must know the costs of resources. This poses the two fundamental questions for ABC that centres around a resource:-

1. What unit resource costs are associated with a resource?

2. How does one deduce unit resource cost(s) so that direct, indirect and overhead costs are accounted for within the costs of a resource?

The solving of these two fundamental problems relevant to ABC has been pivotal towards the research contribution of this report. The goal, scope, direction and agenda towards solving the two fundamental problems with ABC was set by the thesis dissertation statement.

The thesis dissertation statement was formulated after a critical examination of cost literature pertaining to traditional costing, ABC, other cost models such as Target Costing and Throughput Accounting, ABC developments and its current state of art, ABM software, artificial intelligence (AI) and information technology (IT) developments, enterprise modelling and integration effort; and through interactions with several companies from the manufacturing, accounting, consulting and service sectors.

The holistic solution presented in this research towards solving the two fundamental problems of ABC consists of 3 interdependent parts:-

1. *A Cost Ontology for Enterprise Modelling* that promotes a precise and unambiguous understanding of temporal cost behaviour of resources and activities for a formalization of the ABC Principle. It is based upon the TOronto Virtual Enterprise (TOVE) Model that is founded upon the TOVE Core Ontology for activity, state, resource and time. This part identifies and associates costs for a resource and their effects on the cost of an activity requiring the resource.

2. A *Theory of Resource Cost Units* that provides a logical and formalized framework to deduce the basic unit costs of a resource identified in 1, so that conventional concepts of direct, indirect and overhead costs are accounted for in the unit costs of a resource.

3. A *Prototypical Demonstration of a Cost Advisor* that encapsulates axioms, principles and theories of the two previous parts in an IT environment referred to as the TOVE Testbed. It provides a visualization of the "activity-based" enterprise modelling infra-structure and a work-bench to deduce solutions to some cost

related queries, referred to as informal competency questions, as posed by our corporate partners - deHavilland Inc. (Bombardier), Toronto; Broken Hill Proprietary (BHP), Melbourne, Australia.

The Ontological Engineering Methodology taken towards enterprise modelling and integration, and the specification of formal models as logical models found in Computer Science that were undertaken in the development of the solution, enabled the following models to be developed that support the thesis:-

1. a formal, re-usable, descriptive model of enterprise costs, called a Cost Ontology for Enterprise Modelling in TOVE;
2. a formal, re-usable, prescriptive model called a Micro-theory for ABC;
3. a formal, re-usable, descriptive model called a Formalization of Overheads;
4. a formal, re-usable, prescriptive model called a Micro-theory of Resource Cost Units.

This thesis is supported by the design, analysis, and prototypical implementation of the Cost Ontology for Enterprise Modelling, a Micro-theory for ABC and a Micro-theory of Resource Cost Units in the Cost Advisor. The design, analysis, and implementation of the solution support the thesis in the following ways:-

- It shows that costs within an organization can be described by representing them in an enterprise model centered around activity, state, resource, causality and time formalizations or the Core Ontologies in TOVE. This descriptive model of costs, called the Cost Ontology for Enterprise Modelling in TOVE exposes the temporal, granular, and interdependent behaviour of costs due to activities and resource in an enterprise. This provides a Formal Model or Formalization of ABC, and a Micro-theory of ABC.
- It demonstrates that the engineering of ontologies to meet the criteria of competence to answer common sense questions in the cost domain, and re-usability of ontologies to other domains, provides the deductive capability for the

objective prescription of answers to questions about costs with the Micro-theory of ABC.

- It shows a logical extension of the activity and resource ontology in TOVE from a cost perspective towards the Formalization of Overheads so that all costs, be they direct, indirect or overheads costs can be described by representing them in an enterprise model.
- It shows the re-use of the Micro-theory of ABC towards developing a Micro-theory of Resource Cost Units so that costs of a resource can be objectively deduced to include direct, indirect and overhead costs.

In conclusion, the answer to the two fundamental problems with ABC are as follows:-

1. **What unit resource costs are associated with a resource?**

Answer: The unit resource costs associated with a resource are the committed resource cost unit, the enabled resource cost unit, the disenabled resource cost unit, and the reenabled resource cost unit.

2. **How does one deduce unit resource costs so that direct, indirect, and overhead costs are accounted for within the costs of a resource?**

Answer: This deduction can be made by first, describing and representing costs within an enterprise with the Cost Ontology for Enterprise Modelling in TOVE and the Formalization of Overheads; second, by deploying the Micro-theory of Resource Cost Units and the Micro-theory of ABC as presented in this research.

9.2 Contributions

Developments in this report make contributions to the domains of ABC, enterprise modelling, and strategic cost management with ABC (aka ABM) based upon knowledge representations for information systems. Specific contributions are:-

1. This research provides the logical framework to automate and to objectively deduce costs of the resources of an enterprise to include direct, indirect and overhead costs.

2. The Theory of Resource Cost Units promotes direct traceability and accountability of costs - direct, indirect, and overheads, at "point of absorption". It eliminates the confusing and difficult issues that surround the distribution of costs from "cost pools" to activities via "drivers". Moreover, the distribution of overhead costs to activities is cost sensitive to idle plant capacity, i.e., product costs increase due to the higher fixed overhead components (e.g. depreciation, property taxes, leases, cost of capital, etc.) that must be borne by the products due to low plant utilization.

3. The granular temporal cost behaviour and the inter-dependence of costs for an activity and its resources bring the following cost sensitivities to management's attention - costs of resource downtime, costs of activity suspension, costs of resource non-utilization due to it being committed to an activity, and costs of reenabling a resource. This is invaluable to management seeking opportunities for capital investments.

4. This research has been based upon "formal models" of development. This contributes to clearer and less ambiguous implementations for ABC

5. The Principle of Resource Probing provides the trace of in-house processes for the supply of internal resources. This trace is provided in the form of the Commit Envelope of activities. The trace of processes, together with costs of processes to be inclusive of direct, indirect, and overhead costs should prove invaluable to management for "make or buy" decisions regarding resources; and the re-engineering of processes within the supply-chain of internal resources.

6. The Principle of Activity Probing enables one to deduce costs of specifying an activity. It enables companies to focus on areas such as research and development, engineering, human resource planning, advertising and marketing. ABC developments to date have not given sufficient coverage to the costs of specifying an activity.

7. The Cost Advisor serves as a prototype for a new generation of ABM software development - the answering of informal cost queries based upon deductive

reasoning from an AI perspective. The Cost Advisor has been founded upon formal axioms for deduction purposes, a precise and unambiguous representation of activities, states, resources, costs, and "conventional overheads" that form the knowledge base for the enterprise.

8. This research contributes towards taking corporations closer to achieving "open book accounting" [Foster 96].

9. The formal models of this research may serve to narrow the gap and promote a better synergy between Chartered Accountants and Operations' Executives of a corporation. Ground breaking efforts and commitments are already well entrenched into ontological developments by companies such as PricewaterhouseCoopers in SAVILE and COMET (refer to Sec. 2.13 of Ch. 2) towards "formalization" of their accounting and audit practices. The formal models of this research can serve as the "rigid underlying models" that supply the thousands of numbers from the operations side of the business for the "repositories" of accountants to build their "financial models" that consist of the few numbers that are typically the summations of thousands of numbers.

9.3 Future Research

Based upon the groundwork established in this research, some areas for future research may be sought in an attempt to provide solutions to the following:

- How may the need of inventory valuation defined by GAAP in manufacturing organizations be satisfied?
- How may a general ledger be generated from the ABC costs deduced from this research? Recall, that presently cost pools for ABC are sought from the general ledger and then distributed and allocated to activities via drivers. Given that resource costs can be deduced based upon the contributions of this research, can the general ledger be "formally deduced" from ABC costs obtained from the Micro-theory of ABC?

- What links are necessary and how may the contributions be extended to provide historical cost information to serve government regulatory agencies such as Revenue Canada and the Internal Revenue Service of USA?
- What are the ABC cost solutions necessary through the current contributions and their extensions so that quarterly and annual reports of corporations may be generated to be understood by stockholders, financial analysts and potential buyers of a company?

Future research in the above areas may shed light towards some formalized extensions of this research into "Financial Accounting with Enterprise Activity Based Cost Modelling".

References

1. [Albright 95]
   Albright, Thomas L., "Software for Activity-Based Management", Journal of Cost Management, pp. 6-25, Spring 1995.

2. [Allen 83]
   Allen, James F., "Maintaining Knowledge about Temporal Intervals", David Waltz (editor), *Communications of the ACM (Research Contributions),* Vol. 26, No. 11, pp. 832 - 843, 1983.

3. [Babad & Balachandran 93]
   Babad, Yair M. and Balachandran, Bala V., "Cost Driver Optimization in Activity-Based Costing", *The Accounting Review,* Vol. 68, No.3, pp. 563 - 575, July 1993.

4. [Bailey et al. 85]
   Bailey, A. D., Duke, G. L., Gerlach, C., Ko, C., Mersevy, R. D., and Whinston, A. B., "TICOM and the analysis of internal controls", The Accounting Review, LX(2): pp.186 - 201, April, 1985.

5. [Bailey et al. 89]
   Bailey, A. D., Han, K. S., Stansifer, R. D., and Whinston, A. B., "The advanced internal accounting control model using a logic programming approach", Working Paper of 7/20/89, PricewaterhouseCoopers, 1989.

6. [Bailey et al. 90]
   Bailey, A. D., Kiang, Y., Kuipers, B., and Whinston, A. B., "Analytical review and qualitative and causal reasoning in auditing, Internal Working Paper of PricewaterhouseCoopers, Draft of 4/15, 1990.

7. [Banker et. al. 93]
   Banker, Rajiv D., Kauffman, Robert J., & Zweig, Dani., "Repository Evaluation of Software Reuse", *IEEE Transactions on Software Engineering*, Vol. 19, No. 4, pp. 379-389, April 1993.

8. [Berliner & Brimson 89]
   Berliner, C. and Brimson, J.A., *Cost Management for Today's Advanced Manufacturing. The CAM-I Conceptual Design,* Harvard Business School Press, 1989.

9. [Blaxill & Hout 91]
Blaxill, M.F. and Hout, T.M., "The Fallacy of the Overhead Quick Fix", Harvard Business Review, pp. 93-101, July-August 1991.

10. [Booth 92]
Booth, Rupert, "The New Competitive Order", Management Accounting, pp.34-36, November, 1992.

11. [Borjesson 97]
Borjesson, Sofia, "A Case Study on Activity-Based Budgeting", *Cost Management*, pp. 7-18, Winter 1997.

12. [Bouwman 83]
Bouwman, M. J., "Human diagnostic reasoning by computer: An illustration from financial analysis", Management Science, 29(6): pp. 653 - 672, June, 1983.

13. [Brachman 85]
Brachman, Ronald J., What IS-A Is and Isn't: An Analysis of Taxonomic Links in Semantic Networks, IEEE Computer, Vol. 16, No. 10, pp. 30-36, 1985.

14. [Bridgeland 90]
Bridgeland, D. M., "Three qualitative simulation extensions for supporting economics models", *Proceedings 6th IEEE Conference on A. I. Applications,* pp. 266 - 273, Santa Barbara, CA., March, 1990.

15. [Brimson 91]
Brimson, James A., *Activity Accounting: An Activity-based Costing Approach*, John Wiley & Sons, Inc., 1991.

16. [Brinker 96]
Brinker, Barry J., (editor), *Emerging Practices in Cost Management - 1996 Edition,* Warren, Gorham & Lamont.

17. [CAM-I:TC 97]
CAM-I Target Cost Core Group, *Target Costing : the next frontier in strategic cost management*, Irwin Professional Publishers., Chicago, 1997.

18. [Cawsey 97]
Cawsey, Alison. *Essence of Artificial Intelligence*, UK: Prentice Hall, 1997.

19. [Campbell & Shapiro 95]
Campbell, A.E. and Shapiro, S.C., "Ontological Mediation: An Overview", *Proceedings of the IJCAI Workshop on Basic Ontological Issues in Knowledge Sharing,* Menlo Park, CA: AAAI Press, 1995.

20. [Chen 76]
Chen, Peter P., "The Entity Relationship Model: Toward a Unified View of Data", *ACM Transactions on Database Systems,* Vol. 1, No. 1, pp. 9-37, 1976.

21. [Clocksin & Mellish 87]
Clocksin, W.F. and Mellish, C.S., *Programming in Prolog (Third, Revised and Extended Edition),* Springer-Verlag, 1987.

22. [CMG - IMA 94]
Cost Management Group, Institute of Management Accountants, "4th Annual Activity-Based Cost Management Survey Results", *Cost Management Update*, pp. 1-4, February 1994.

23. [Cokins 96]
Cokins, Gary, *Activity-Based Cost Management - Making It Work*, The McGraw-Hill Companies, Inc., USA, 1996.

24. [Connolly & Ashworth 94]
Connolly, Tim and Ashworth, Gary, "Activity-Based Management: Software Issues and Options", Management Accounting (UK), pp.34-36, May 1994.

25. [Cooper 88a]
Cooper, R., "The Rise of Activity Based Costing - Part One: What is an Activity-Based Cost System?", *Journal of Cost Management*, pp.45-54, Summer 1988a.

26. [Cooper 88b]
Cooper, R., "The Rise of Activity Based Costing - Part Two: When Do I Need an Activity-Based Cost System?", Journal of Cost Management, pp.41-48, Fall 1988b.

27. [Cooper 90]
Robin Cooper, ABC: A NEED, NOT AN OPTION, Accountancy, pp. 86-88, September, 1990.

28. [Cooper 96]
Cooper, Robin, "Activity-Based Costing and the Lean Enterprise", Cost Management, pp. 6-14, Winter 96.

29. [Cooper et al. 92]
Cooper, R., , Kaplan, Robert S., Maisel, Lawrence S., and Morrisey, Eileen, "From ABC to ABM: Does Activity-based Management Automatically Follow From An Activity-based Costing Project?", *Management Accounting*, pp. 54-57, Nov. 1992.

30. [Cooper & Kaplan 92]
Cooper, Robin and Kaplan, Robert S., "Activity-Based Systems: Measuring the Costs of Resource Usage", *Accounting Horizons*, September 1992.

31. [Cunningham 92]
Cunningham, Gary M., "Management Control and Accounting Systems under a Competitive Strategy", *Accounting Auditing & Accountability Journal,* Vol. 5, No. 2, pp. 85-102, 1992.

32. [Cusamano 91]
Cusamano, M., *Japan's Software Factories: A Challenge to U.S. Management*, Oxford, England: Oxford University Press, 1991.

33. [Darlington et al. 92]
Darlington, John; Innes, John; Mitchell, Falconer; and Woodward, John; "Throughput Accounting: the Garrett Automotive Experienc", *Management Accounting,* pp.32-38, Management Accounting, April 1992.

34. [DoD 93]
The Office of Director of Defense Information, The DoD Enterprise Model, Office of the Secretary of Defense, Washington D.C., Feb. 16, 1993.

35. [Dugdale & Shrimpton 90]
Dugdale, David and Shrimpton, Sue, "Product Costing in a JIT Environment", Management Accounting, pp. 40-42, March 1990.

36. [Elliott 83]
Elliott, R. K., "Unique audit methods: Peat Marwick International", Auditing: A Journal of Practice and Theory, 2(2): pp. 1 -12, Spring, 1983.

37. [ESPRIT 91]
ESPRIT Consortium AMICE, Open System Architecture for CIM, CIM-OSA AD 1.0, Architecture Description, (ESPRIT - Project 688, Project 2422, Project 5288), 2 Boulevard de la Woluwe Bte. 8, B-1150 Brussels, Belgium, 1991.

38. [Fadel et al. 94]
Fadel, F.G., Fox, M.S., and Gruninger, M., "A Generic Enterprise Resource Ontology", *Proceedings of Third Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises* (WET-ICE), Morgantown, WV, pp. 117-128, April, 1994.

39. [Foster 96]
Foster, George, "Management Accounting in 2000", *Cost Management*, pp.36-39, Winter 1996.

40. [Fox 92]
Fox, M.S., "The TOVE Project, Towards a Common Sense Model of the Enterprise", in *Enterprise Integration*, Ch. Petrie (Ed), Cambridge, MA, MIT Press, 1992.

41. [Fox 90]
Fox, M. S., "Artificial Intelligence and Expert Systems: Myths, Legends, and Facts", *IEEE Expert*, 1990.

42. [Fox et al. 93]
Fox, Mark S., Chionglo, John F., Fadel, Fadi G., "Towards Common Sense Modelling of an Enterprise", *Proceedings of the Second Industrial Engineering Research Conference,* Norcross, GA., pp. 425-429, 1993.

43. [Fox et al. 95]
Fox, M.S., Barbuceanu, M., and Gruninger, M., "An Organisation Ontology for Enterprise Modelling: Preliminary Concepts for Linking Structure and Behaviour", *pp. 71-81, Proceedings of Fourth Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises* (WET-ICE), Berkeley Springs, WV, April, 1995.

44. [Gruninger & Fox 94a]
Gruninger, M. and Fox, M.S., "An Activity Ontology for Enterprise Modelling", *Proceedings of Third IEEE Workshop on Enabling Technologies: Infrastructures for Collaborative Enterprises (WET ICE 94),* Morgantown, WV, 1994.[

45. [Fox & Gruninger 94]
Fox, M.S. and Gruninger, M., "Enterprise Engineering: An Information Systems Perspective", *Proceedings of 3rd Industrial Engineering Research Conference*, Atlanta, Georgia, 1994.

46. [Fox & Gruninger 96]
Fox, M.S., Gruninger, M., *Enterprise Modelling*, Department of Mechanical and Industrial Engineering, University of Toronto, 4 Taddle Creek Rd., Toronto, Ontario, M5S 3G9, 1996.

47. [Fox & Gruninger 97]
Fox, M.S. and Gruninger, M., "Ontologies for Enterprise Modelling", *Proceedings of ICEIMT'97,* International Conference on Enterprise Integration and Modelling Technology, Torino, Italy, October 28 - 30, 1997.

48. [Goldratt 90]
Goldratt, Eliyahu M., *Theory of Constraints*, North River Press, 1990.

49. [Goldratt 90]
Goldratt, Eliyahu M., *The Haystack Syndrome: sifting information out of the data ocean,* North River Press, 1990.

50. [Goldratt & Cox 84]
Goldratt, E., and Cox, J., *The Goal - Excellence in Manufacturing,* North River Press, New York, 1984.

51. [Gruber 93]
Gruber, T.R., "Toward principles for the design of ontologies used for knowledge sharing", *Formal Ontology in Conceptual Analysis and Knowledge Representation (N. Guarino and R. Poli, editors),* Kluwer Academic Publishers, 1993.

52. [Gruninger 96]
Gruninger, Michael, "Designing and Evaluating Generic Ontologies", *Proceedings of the Workshop on Ontological Engineering*, European Conference on Artificial Intelligence, Budapest, pp.53-65, 1996.

53. [Gruninger 97]
Gruninger, Michael, "Integrated Ontologies for Enterprise Modelling", *Proceedings of ICEIMT'97,* International Conference on Enterprise Integration and Modelling Technology, Torino, Italy, October 28 - 30, 1997.

54. [Gruninger & Pinto 95]
Gruninger, M. and Pinto, Javier A., "A Theory of Complex Action for Enterprise Modelling", *Working Notes AA AI Spring Symposium Series 1995: Extending Theories of Action: Formal Theory and Practical Applications,* Stanford, CA., U.S.A.

55. [Guha et al. 90]
Guha, R.V., Lenat, D.B., Pittman, K., Pratt, D., and Shepherd, M., "Cyc: A Midterm Report", Communications of the ACM, Vol. 33, no. 8, August 1990.

56. [Guida 94]
Guida, Giovanni, Design and development of knowledge-based systems: from life cycle to methodology, Chichester; New York, 1994.

57. [Hamscher 91]
Hamscher, Walter C., "Model-based financial data interpretation", *Working Notes of the 2nd International Workshop on Principles of Diagnosis* (Technical Report RT/D1/91-10-7, Dipartimento di Informatica, Universita di Torino, 1991), *Proceedings of 1st. International Conference on AI Applications on Wall Street,* New York, (IEEE Computer Society Press), October, 1991.

58. [Hamscher 94]
Hamscher, Walter C., "Modeling accounting systems to support multiple tasks: A progress report", Working Paper of PricewaterhouseCoopers, Price Waterhouse Technology Centre, Menlo Park, CA, 1994.

59. [Hart et al. 86]
Hart, P. E., Barzilay, A., and Duda, R. O., "Qualitative reasoning for financial assessments: A prospectus", *AI Magazine,* 7(1): pp. 62 - 68, Winter, 1986.

60. [Hawks et al. 92]
Val D. Hawks, Val D., Bonadies, Michael P., A. Strong, Brent, and Reid Michael C., "Identifying Costs in Electronics Manufacturing Using Activity-based Costing", *The Journal of Applied Manufacturing Systems*, Vol. 5, No. 2, pp. 70- 81, Winter 1992.

61. [ICAM 81]
Integrated Computer Aided Manufacturing (ICAM), Architecture Part II, Vol. V - Information Modeling Manual (IDEF), AFWAL-TR-81-4023, Wright-Patterson Air Force Base, Ohio, 45433, June 1981.

62. [Innis & Mitchell 91]
Innes, J. & Mitchell, F., Activity Based Costing - A Review with Case Studies, The Chartered Institute of Management Accountants, 1991.

63. [Jayson 87]
Jayson, S., "Goldratt and Cox: revolutionising the factor floor", *Management Accounting* (USA), pp.18-22, May 1987.

64. [Jeans & Morrow 89]
Mike Jeans, Mike and Morrow, Michael, "The practicalities of using activity-based costing", Management Accounting, pp.25-28, November 1989.

65. [Johnson 88]
Johnson, H.T., "Activity Based Information: A Blueprint for World-Class Management Accounting", *Management Accounting (USA)*, pp. 23-30, June 1988.

66. [Johnson 92]
Johnson, H. Thomas, "It's time to Stop Overselling Activity-Based Concepts", *Management Accounting,* September, 1992.

67. [Johnson & Kaplan 87]
Johnson, H. Thomas and Kaplan, Robert S., *Relevance Lost, The Rise and Fall of Management Accounting*, Boston: Harvard Business School Press, 1987.

68. [Jorysz & Vernadat 90]
Jorysz, H. R. and Vernadat, F. B., "CIM-OSA Part 1: total enterprise modelling and function view", and "CIM-OSA Part 2: information view", *International Journal of Computer Integrated Manufacturing,* Vol. 3, Nos. 3 and 4, pp. 144 - 167, 1990.

69. [Kaplan 88]
Kaplan, Robert S., "One Cost System Isn't Enough", *Harvard Business Review,* pp.61-66, January-February, 1988.

70. [Kaplan & Norton 92]
Kaplan, Robert S. and Norton, David P., "The Balanced Scorecard - Measures That Drive Performance", Harvard Business Review, pp. 71-79, January-February, 1992.

71. [Klittich 90]
Klittich, M., "CIM-OSA Part 3: CIM-OSA integrating infrastructure - the operational basis for integrated manufacturing systems", *International Journal of Computer Integrated Manufacturing*, Vol. 3, Nos. 3 and 4, pp. 168 - 180, 1990.

72. [Konsynski & McFarlan 90]
Konsynski, Benn R. and McFarlan, E. Warren, "Information Partnerships - Shared Data, Shared Scale", Harvard Business Review, pp. 114 - 120, Sept.-Oct. 1990.

73. [Kosanke 91]
Kosanke, K., CIM-OSA: Its Role in Manufacturing Control, ESPRIT Consortium AMICE, Bruxelles, Belgium, 1991.

74. [Kosanke 97]
Kosanke, K., "Enterprise Integration - International Consensus: A Europe - USA Initiative", *Proceedings of ICEIMT'97,* International Conference on Enterprise Integration and Modelling Technology, Torino, Italy, October 28 - 30, 1997.

75. [Ladkin 86]
Ladkin, P., "Primitives and Units for Time Specification", Kestrel Institute, 1801 Page Mill Road, Palo Alto, Ca 94304-1216, *Science: Knowledge Representation,* pp. 354 -359, 1986.

76. [Malcolm 91]
Malcolm R., "Overhead Control Implications of Activity Costing", *Accounting Horizons,* pp. 69 - 78, December 1991.

77. [McCarthy & Hayes 69]
McCarthy, J., & Hayes, P. J., "Some Philosophical Problems from the Standpoint of AI", in *Machine Intelligence 4*, Meltzer B., Michie D.(eds.), Edinburgh,UK:Edinburgh University Press, pp. 463-502, 1969.

78. [Miller & Vollman 85]
Miller, J.G. and Vollman, T.E., "The Hiddent Factory", *Harvard Business Review,* pp. 142 - 150, September/October, 1985.

79. [Monden & Hamada 91]
Monden, Yasuhiro, and Hamada, Kazuki, "Target Costing and Kaizen Costing in Japanese Automobile Companies", Journal of Management Accounting Research, Volume Three, pp. 16-34, Fall 1991.

80. [Morgan & Weerakoon 89]
Morgan Malcolm J. and Weerakoon, Prasanna S. H., "Japanese management accounting: its contribution to the Japanese economic miracle", *Management Accounting,* pp. 40 - 43, June 1989.

81. [Nado et al. 91]
Nado, R., Van Baalen, J., and Fikes, R., "JOSIE: An integration of specialized representation and reasoning tools", *ACM Sigart Bulletin special issue on implemented knowledge representation and reasoning systems,* 2(3): pp. 101 - 107, June, 1991.

82. [Nicholls 92]
Nicholls, Brent, "ABC in the UK - A Status Report", *Management Accounting*, May, 1992.

83. [O'Brien 90]
O'Brien, Thomas, "Improving Performance Through Activity Analysis", *Performance Excellence in Manufacturing and Service Organizations*, Peter B.B. Turney, editor, (Sarasota: American Accounting Association), 1990

84. [Piper & Walley 90]
Piper, J.A. and Walley, P., "Testing ABC Logic", *Management Accounting*, pp.37-42, September 1990.

85. [Plossl 90]
Plossl, George W., "Cost accounting in manufacturing: dawn of a new era", *Production Planning & Control*, Vol. 1, No. 1, pp. 61-68, 1990.

86. [Porter 80]
Porter, Michael E., *Competitive Strategy,* The Free Press, New York, 1980.

87. [Porter 85]
Porter, Michael E. , *Competitive Advantage,* The Free Press, New York, 1985.

88. [Plossl & Wight 67]
Plossl, G.W. and Wight, O. W., Production and Inventory Control - Principles and Techniques, Prentice Hall, 1967.

89. [Quintus 91]
Quintus Corporation, An Integration Co., *Quintus X Window Interface*, July, 1991.

90. [Radhakrishnan & Srinidhi 97]
Radhakrishnan, Suresh and Srinidhi, Bin, "Avoiding the Death Spriral: A Case for Activity-Based Costing", *Cost Management*, pp. 19-24, Winter 1997.

91. [Roberts & Silvester 96]
Roberts, Micheal W. and Silvester, Katherine J., "Why ABC Failed and How It May Yet Succeed", *Cost Management*, pp. 23-35, Winter 1996.

92. [ROCK 91]
ROCK Software Functional Specification, Carnegie Group, Inc., Version 2.0, October 1991.

93. [Ross 77]
Ross, D.T., "Structured Analysis (SA): A Language for Communicating Ideas", *IEEE Transactions on Software Engineering,* SE-3, pp.16-24, 1977.

94. [Rotch et al. 91]
Rotch, William, Allen Brandt R. and Smith, C. Ray, *The Executive's Guide to Management Accounting and Control Systems,* Fourth Edition, Dame Publications, Inc., 1991.

95. [Ruhl 97]
Ruhl, J. M., "Activity-Based Cost Management: Making It Work", *Journal of Cost Management,* Volume 11, No.1, pp.46-47, January/February, 1997.

96. [Sakurai 89]
Sakurai, M., "Target Costing and How to Use It", *Journal of Cost Management*, pp.39-50, Summer 1989.

97. [Scheer 85]
Scheer, August-Wilhelm, *Enterprise-Wide Data Modelling,* Springer-Verlag, 1985.

98. [Scheer 88]
Scheer, August-Wilhelm, *CIM Computer Integrated Manufacturing Computer Steered Industry,* Springer-Verlag, 1988.

99. [Scheer 89]
Scheer, August-Wilhelm, *Computer: A Challenge for Business Administration*, Springer-Verlag, 1989.

100. [Scheer 91]
Scheer, August-Wilhelm, *CIM Towards the Factory of the Future,* Second, Revised and Enlarged Edition, Springer-Verlag, 1991.

101. [Schonberger 86]
Schonberger, Richard J., *World Class Manufacturing: Lessons of Simplicity Applied,* pp. 178-180, The Free Press, New York, 1986.

102. [Shepperd and Schofield 96]
Shepperd, Martin and Schofield, Chris, *"Effort Estimation Using Analogy"*, Proceedings of ICSE 18, Berlin 1996

103. [Shields & McEwen 96]
Shields, Michael D., and McEwen, Michael A., "Implementing Activity-Based Costing Systems Successfully", Cost Management, pp.15-22, Winter 1996.

104. [SMA 93]
Society of Management Accountants of Canada, "Activity-Based Costing, Management Accounting Issues Paper 3", *Issues Papers*, SMA, Hamilton, Ontario, Canada, L8N 3C3, 1993.

105. [Sheridan 89]
Sheridan, Thomas, "Don't Count Your Costs - Manage Them", *Management Accounting*, pp. 20 - 24, February 1989.

106. [Stoffel 92]
Stoffel, Thomas J., "Activity-based Costing: The Competitive Advantage for the 1990s", *The Journal of Applied Manufacturing Systems,* pp. 58-63, Winter 1992.

107. [Shank & Govindrajan 88]
Shank, J. K. and Govindrajan, V., "The Perils of Cost Allocation Based on Production Volumes", *Accounting Horizons,* pp. 71-79, December 1988.

108. [Tham 93]
Tham, K. Donald, "Cost Perspectives in Enterprise Models", *Working Paper for the Enterprise Integration Laboratory (EIL)*, Dept. of Industrial Engineering, U of T, September 1993.

109. [Tham et al. 94]
Tham, K. Donald, Fox, M.S., and Gruninger, M., "A Cost Ontology for Enterprise Modelling", *Proceedings of Third Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises* (WET-ICE), Morgantown, WV, pp. 197-210, April, 1994.

110. [Tham & Fox 98]
Tham, K. Donald and Fox, Mark S., "Enterprise Models and Their Cost Perspectives", *Proceedings for Industrial Engineering and Management*, Canadian Society for Mechanical Engineering (CSME) Forum, Toronto, May 19-22, 1998.

111. [TOVE 93]
TOVE Working Paper, Enterprise Integration Laboratory, January 1993.

112. [Turney 92(A)]
Turney, Peter B.B., *Common Cents: The ABC Performance Breakthrough*, Portland, OR: Cost Technology, 1992.

113. [Turney 92(B)]
Turney, Peter B.B., "Activity-based Management", *The Journal of Applied Manufacturing Systems*, pp. 29-36, Winter 1992.

114. [Ullman 88]
Ullman, Jeffrey D., Principles of Database & Knowledge Systems, Computer Science Presses, 1988.

115. [Tai 92]
Tai, Shuji, "Managerial Accounting and the 'New Managerial Ideas' - A Case Study of the Matsushita Electric Industrial Co. Ltd", *Accounting Auditing & Accountability Journal*, Vol. 5, No. 4, pp. 3-34, 1992.

116. [Tatikonda & Tatikonda 91]
Tatikonda, Lakshmi U., & Tatikonda, Rao J., "Overhead Cost Control - Through Allocation or Elimination?", *Production and Inventory Management Journal*, pp. 37- 41, 1st Quarter 1991.

117. [Vernadat 97]
Vernadat, F.B., "Enterprise Modelling Languages", *Proceedings of ICEIMT'97*, International Conference on Enterprise Integration and Modeling Technology, Torino, Italy, October 28-30, 1997.

118. [Williams 92]
Williams, Theodore J., *The Purdue Enterprise Reference Architecture*, Purdue Laboratory for Applied Industrial Control, Purdue University, West Lafayette, Indiana 47907, USA, March 16, 1992.

119. [Williams 97]
Williams, T.J., "Methodologies for Enterprise Integration", *Proceedings of ICEIMT'97,* International Conference on Enterprise Integration and Modelling Technology, Torino, Italy, October 28 - 30, 1997.

120. [Zeidenberg 92 (Fox - Globe)]
Zeidenberg, Jerry, "Creating Common-Sense Systems", *The Globe and Mail,* March 3, 1992.

121. [Zhan 94]
Zhan, Yin, *A Users' Guide for Oak, Version 0.5.3.*, Enterprise Integration Laboratory, Dept. of Industrial Engineering, University of Toronto, Sept. 16, 1994.

122. [Zhuang & Burns 92]
Zhuang, Lizhang, and Burns, George, "Activity-based Costing in Non-standard Route Manufacturing", *International Journal of Operations & Production Management*, Vol. 12, No. 3, pp. 38 - 60, 1992.

Glossary

A

activity accounting
: the collection of financial and operational performance information about significant activities of the business

activity driver
: a factor used to assign cost from an activity tp a cost object

advisor
: A software tool which encapsulates, and enables performing tasks using, ontologies and micro-theories. The tasks for an advisor fall into evaluation, analysis, and guidance.

analysis capability of an advisor
: capability to use an advisor to predict, monitor, identify, and explain phenomena in a domain modelled by ontologies and micro-theories encapsulated by the advisor.

ABC
: acronym for Activity-Based Costing

ABM
: acronym for Activity-Based Management

Artificial Intelligence (AI)
: a branch of computer science that uses computer programs to solve problems that appear to require human deductive reasoning

axiom
: semantics which define, and constrain the use of, the terminology in a model.

B

benchmark
: an activity that is *best practice* and by which a similar activity will be judged

bill of activities
: a list of activities and costs associated with a cost object

C

classification hierarchy
: a taxonomy; a tree structure displaying is-a relationships in a hierarchical manner.

common-sense model
: A model of a domain in which the core, fundamental concepts of that domain are represented such that the model is able to answer common sense questions by means of deduction.

competency
The capability of a set of representations to support problem solving within one or more domains.

competency question
A question which characterizes the problem solving or task support capability of an ontology or micro-theory. It must be possible to answer a competecy question using the representations of the ontology or micro-theory that it motivate

constraining axioms
First-order logic sentences that constrain the interpretation upon primitive terms and definitions.

conventional (or traditional) cost system
any of the older, traditional cost systems that use direct material and labour consumed as the primary means of apportioning overheads

cost
entity that represents the temporal fiscal or monetary dimension, attribute, or characteristic of an enterprise activity, and may be referred to as activity cost

cost driver
a factor that causes a change in the performance of an activity and, in doing so, affects the resources required by the activity

D

definitions (in an ontology)
Formalization in first-order logic of terminology already introduced; these are defined in terms, of existing terms that have been previously defined.

descriptive model
A model that describes the characteristics of a domain.

domain value
An allowable value for an attribute of an object.

E

enterprise engineering
The process of using an enterprise model to analyze and design an enterprise; the process of engineering an enterprise by manipulating a model of that enterprise.

enterprise model
A computational representation of the structure, activities, processes, information, resources, people, behaviour, goals, and constraints of a business, govenment, or other enterprise.

evaluation capability of an advisor
the capability of an advisor to compare different models along a dimension, such as quality or cost, and to evaluate that one model is better as per that dimension.

F

first-order logic
: A formal language which supports expressing propositions as well as predicates, where predicates may have quantified variables as arguments. In Higher-order logic, the predicates themselves are quantified.

formal logic
: A language with restrictive syntax and semantics, which supports deduction or inference of propositions given initial propositions and axioms.

formal model
: A data model expressed using formal logic.

formalization
: Expressions, written in the logic-based language, of the semantics of a model's terminology.

G

ground Terms
: Primitive terms instantiated with facts.

guidance capability of an advisor
: the capability of an advisor to suggest alternatives based upon deductions using ontologies and micro-theories it encapsulates.

I

instantiated model
: A formal model populated with data.

M

micro-theory
: A formal model required to solve a problem in a domain or to describe in detail a subset of the domain; a contextually-bounded formal model of knowledge that is often task-oriented.

minimal ontological commitment
: A design guideline to restrict axioms of an ontology to those required to minimally describe a domain. Thus the ontology offers only minimal commitment to give details or facilitate problem solving about the domain.

O

object
: A collection of entities organized as one because they share common properties.

ontology
: A formal description of entities and their properties; it forms a shared terminology for

objects of interest in the domain, along with definition for the meaning of each of the terms.

P

predicate
: An expression of an object or a relationship between objects in a formal language. In first-order logic, predicates have one or more arguments, where all variable (non-constant) arguments must be quantified.

predicate calculus
: See first-order logic (Appendix B)

predicate logic
: See first-order logic (Appendix B)

prescriptive model
: A model that prescribes an alternate state for a domain.

primitive term
: Predicates which are never formally defined in first-order logic.

Q

queries
: A question, expressed in the formal language, that initiates a deduction about the model.

R

reducibility
: Demonstration that competency questions of another model- spanning a different competency than the evaluated model- can be reasonably translated (reduced) to competency questions answerable using representations of the evaluated model.

relation
: Relationships between objects in a data model.

representations
: Informal statements of terminology and semantics, e.g., in English, as well as formalizations of a model.

re-usability
: Capability to use portions of a model to solve different problems beyond the problems that initially motivated the development of the model.

S

semantics
: Meaning of the terminology of a model.

spans competency
: A native ontology spans the competency of a target ontology if a set of the target ontology's competency questions can be reduced to competency questions expressible and answerable using target ontology representations.

syntax
: The grammar for composing expressions in a model.

T

taxonomy
: a tree structure displaying is-a relationships in a hierarchical manner

terminology
: Terms in the vocabulary of a model.

Appendix A: List of Symbols

Object-oriented modelling:

a:	activity
r:	resource
s:	state
t:	time
c:	cost
v:	resource cost unit

Situation Calculus:

σ:	extant or hypothetical situation
f:	fluent

First Order Logic (FOL) axioms:

$\forall$:	for all
$\exists$:	there exists
$\wedge$:	and
$\vee$:	or
$\neg$:	not,
$\supset$:	(if ... then)
$\equiv$:	(if and only if)

Set Theory:

$\cap$:	intersection of 2 sets
$\cup$:	union of 2 sets

Appendix B: Primer on First Order Logic

(Reproduced by permission, courtesy of Henry Kim, PhD Candidate, U of T)

FOL.1 Introduction

In this dissertation, first-order predicate logic[1] is used as the mathematical language with which an enterprise is modelled. This chapter is provided to offer the dissertation reader a brief primer for understanding the terms and concepts represented in first-order logic. This primer is excerpted from [Cawsey 97].

Predicate logic allows us to represent fairly complex facts about the world, and to derive new facts in a way that guarantees that, if the initial facts were true then so are the conclusions. It is a well understood formal language, with well-defined syntax, semantics and rules of inference.

FOL.2 Propositional Logic

Predicate logic is a development of *propositional* logic. In proposition logic a fact such as "Alison likes waffles" would be represented as a simple atomic proposition. Lets call it P. We can build up more complex expressions (*sentences*) by combining atomic propositions with the *logical connectives* $\wedge$ (and), $\vee$ (or), $\neg$ (not), $\supset$ (if ... then), and $\equiv$ (if and only if). So, if we had the proposition Q representing the fact "Alison eats waffles," we could have the facts:

$P \wedge Q$ "Alison likes waffles <u>and</u> Alison eats waffles."

$P \vee Q$ "Alison likes waffles <u>or</u> Alison eats waffles."

$\neg Q$ "Alison does <u>not</u> eat waffles."

$P \supset Q$ "<u>If</u> Alison likes waffles <u>then</u> Alison eats waffles."

$P \equiv Q$ "Alison likes waffles <u>if and only if</u> Alison eats waffles. That is, "if Alison likes waffles then Alison eats waffles, and if Alison eats waffles then Alison likes waffles."

1. In this dissertation, the terms "first-order logic," "first-order predicate logic," "predicate logic," and "predicate calculus" are used interchangeably.

In general, if X and Y are sentences in propositional logic, then so are $X \wedge Y$, $X \vee Y$, $\neg X$, $X \supset Y$, and $X \equiv Y$. So the following are valid sentences in the logic:

$P \vee \neg Q$.
$P \wedge (P \supset Q)$.
$(Q \vee \neg R) \supset P$.

Propositions can be true or false in the world. An *intepretation* function assigns, to each proposition, a truth value (i.e., true or false). This interpretation function says what is true in the world. We can determine the truth value of arbitrary sentences using *truth tables* which define the truth values of sentences with logical connectives in terms of the truth values of their component sentences. The truth tables provide a simple *semantics* for expressions in propositional logic. As sentences can only be true or false, truth tables are very simple, for example:

X	Y	$X \wedge Y$
T	T	T
T	F	F
F	T	F
F	F	T

In order to infer new facts in a logic we need to apply *inference rules*. The semantics of the logic will define which inference rules are universally *valid*. One useful inference rule is the following (called modus ponens) but many others are possible:

$a, a \wedge b$.
$\therefore b$.

The above rule just says that if $a \wedge b$ is true, and a is true, then b is necessarily true. We could prove that this rule is valid using truth tables.

FOL.3 Predicate Logic: Syntax

The trouble with propositional logic is that it is not possible to write general statements in it, such as "Alison eats everything that she likes." We'd have to have lots of rules, for every different thing that Alison liked. Predicate logic makes such general statements possible.

Sentences in predicate logic are built up from *atomic sentences*, which consist of a predicate name followed by a number of arguments. These arguments may be any *term*. Terms may be:

Constant symbols, such as 'alison.'
Variable symbols, such as 'X.'
Function expressions, such as 'father(alison).'

So, atomic sentences in predicate logic include the following:

friends(alison, richard).
friends(father(fred), father(joe)).
likes(X, richard).

Sentences in predicate logic are constructed (much as in propositional logic) by combining atomic sentences with logical connectives, as in the following:

friends(alison, richard) ⊃ likes(alison, richard).
likes(alison, richard) ∨ likes(alison, waffles).
((likes(alison, richard) ∨ likes(alison, waffles)) ∧ ¬likes(alison, waffles)) ⊃ likes(alison, richard).

Sentences can also be formed using *quantifiers* to indicate how any variables in the sentence are to be treated. The two quantifiers in predicate logic are ∀ (for all) and ∃ (there exists), so the following are valid sentences:

∃X bird(X) ∧ ¬flies(X). "There exists some bird that doesn't fly."
∀X (person(X) ⊃ ∃Yloves(X,Y)). "Every person has something they love."

A sentence should have all its variables quantified. So strictly, an expression like "∀X loves(X, Y)," though a well formed *formula* of predicate logic, is not a sentence.

FOL.4 Predicate Logic: Semantics

The semantics of predicate logic is defined (as in propositional logic) in terms of the truth values of sentences. Like in propositional logic, we can determine the truth value of any sentence in predicate logic if we know the truth values of the basic components of that sentence. An *interpretation* function defines the basic meanings/truth values of the basic components, given some *domain* of objects that we are concerned with.

In propositional logic we saw that this interpretation function was very simple, just assigning truth values to propositions. However, in predicate calculus we have to deal with predicates, variables and quantifiers, so things get much more complex.

Predicates are dealt with in the following way. If we have, say, a predicate P with 2 arguments, then the meaning of that predicate is defined in terms of a mapping from all possible pairs of objects in the domain to a truth value. So, suppose we have a domain with just three objects in: 'fred,' 'jim,' and 'joe.' We can define the meaning of the predicate 'father' in terms of all the pairs of objects for which the 'father' relationship is true —say 'fred' and 'jim.'

The meaning of ∀ and ∃ are defined again in terms of the set of objects in the domain. "∀X S." means that for every object X in the domain, S is true. "∃X S." means that for some object X in the domain, S is true. So, "∀X father(fred,X).", given our world (domain) of 3 objects ('fred,' 'jim,' 'joe'), would only be true if "father(fred,X)" was true for each object. In our interpretation of the 'father' relation this only holds for "X=jim," so the whole quantified expression will be false in this interpretation.

The term "first-order" refers to the scope of quantification. In a first-order predicate logic sentence, only variables are quantified. In a "higher-order" predicate logic sentence, the predicates themselves are quantified.

FOL.5 Proving Things in Predicate Logic

To prove things in predicate logic we need two things. First we need to know what inference rules are valid—we can't keep going back to the formal semantics when trying to draw a simple inference! Second we need to know a good proof procedure that will allow us to prove things with the inference rules in an efficient manner.

When discussing propositional logic we noted that a much used inference rule was modus ponens. This rule is a *sound* rule of inference for predicate logic. Given the semantics of the logic, if the premises are true then the conclusions are guaranteed true. Other sound inference rules include *modus tollens* (if A⊃B is true and B is false then conclude ¬A), *and-elimination* (if A∧B is true then conclude both A is true and B is true), and lots more.

In predicate logic we need to consider how to apply these rules if the expressions involved. For example we would like to be able to use the facts "∀X (man(X) ⊃ mortal(X))." and "man(socrates)." and conclude "mortal(socrates)." To do this we can use modus ponens, but allow universally quantified sentences to be *matched* with other sentences. So, if we have a sentence "∀X A ⊃ B." and a sentence C then if A and C can be matched or unified then we can apply modus ponens.

The best known general proof procedure for predicate calculus is *resolution*. Resolution is a sound proof procedure for proving things by refutation—if you can derive a contradiction from ¬P then P must be true. Rules for resolution are applied to create a systematic proof procedure. So, under certain conditions, it is possible to automatically prove the truth of a given sentence. The Prolog programming language includes such a proof procedure.

FOL.6 Examples

These are some example first-order predicate logic sentences.

∃X tables(X) ∧ ¬numberofleges(X,4). "There is some table that doesn't have 4 legs."

∀ X (macintosh(X) ⊃ ¬realcomputer(X)). "No macintosh is a real computer," or "If something is a macintosh then it is not a real computer."

∀ X glaswegian(X) ⊃ (supports(X,rangers) ∨ supports(X,celtic)). "All Glaswegians support either Celtic or Rangers."

∃ X small(X) ∧ on(X,table). "There is something small on the table."

¬∃X brusselsprouts(X) ∧ tasty(X). "There are no tasty brussel sprouts."