A HYBRID DESIGN OF COST ACCOUNTING INFORMATION SYSTEMS

Jia-Lang Seng

諶 家 蘭 *

摘　要

成本會計資訊系統係由於成本方法、成本流程兩成本分攤等的不同而非常複雜，不易開發簡便、標準及延伸性的係統軟體，更無法與整合製造系統和製造管理資訊系統作結合，造成資訊結構的中空與斷層。本文提出混合式系統設計，以模組階層式方法，整合分析設計三種不同成本系統，產生標準、透通、及再用性高之軟體之件，解決製造業資訊結構完整性和標準化問題。

關鍵詞：成本會計資訊系統，企業資訊系開發，由下至上分析，由上至下設計模組化，延展性，再用，整合製造系統，製造業資訊結構。

Abstract

How to design a simple, standard, and scaleable cost accounting information system, CAIS, is critical to the success of the corporate information systems development. Due to the variety of cost methods, cost bases, and cost procedures, the CAIS proprietary properties have hindered the endeavors to build simple and standard CAIS software packages which can be applicable to a wider corporate users. Little literature is found to completely and thoroughly address this issue nor to provide a formal, systematic, and methodological design approach. Current state-of-the-practice builds CAIS from scratch, outsources the CAIS project, or simply limits the use of CAIS software packages. None of these approaches is satisfactory nor adequate. We present in this paper a study which addresses this design issue and proposes a module-based, structure-open, and method-mixed CAIS which focuses on the data model and the process model plannings as the formal and systematic solution template to build modular and reusable costing systems. Our solution is a hybrid design of CAIS modules. In the design, we create a three-level module hierarchy based on the concepts of bottom-up functionality and top-down commonality. We apply the relational model to develop the entity-centric database schema. In implementation of the data and process modeling, we have built a system prototype on Windows NT platform in order to show the promise of this design approach.

Keywords: Cost Accounting Information System, Corporate Information Systems Development, Bottom-up Analysis, Top-down Design, Modularity, Scalability, Reusability, Manufacturing Management Information Systems, Corporate Information Infrasture, Relational Model.

* 作者爲本校資訊管理學系副教授

Associate Professor, Department of Management Information Systems
National Cheng-Chi University, Mu-Cha, 116, Taipei, Taiwan, ROC

1. Introduction

Major cost accounting systems include the job order cost system, the process cost system, and the standard cost system [1]. These cost systems have been widely used and implemented in the manufacturing industry. In order to have the total computer-integrated manufacturing (CIM) platform, and the seamless synthesis with the general accounting information system (AIS), and the quality management support and control, the corporate users must computerize the cost accounting information systems (CAIS). The computerization has to be simple, standard, and scalable in order to satisfy the demanding and changing requirements of manufacturing management information systems (MMIS).

However, the current state-of-the-art approach is the ad-hoc computer-based cost accounting information systems (CBCAIS) [4][6]. The variety of cost methods and the complexity of the cost computation, such as the different cost factors, the different cost bases, the different cost flows, and the different cost procedures, are the main barriers to the development of the simple and standard CAIS [6][7]. These barriers hinder the enhancement of an integral corporate information infrastructure (CII). Without an adequate CAIS, it is evident that there is no possibility of quality support for the management decision and control. Decisions fail and controls break. CIM cannot be possible and one integral AIS is out of the question. Redundant and inconsistent data could prevail everywhere and at anytime. These problems surely affect the performance and productivity of the business. Incorrect and out-of-date information will end up with wrong decisions, late decisions, and no decisions. Companies will lose the competitive advantages and miss the opportunity to change and grow. The fast and deep ripple effects of the poor CAIS are overwhelming and cannot be ignored. It is crucial to design and develop a formal and systematic CAIS.

As discussed in the above, due to the difficult proprietary nature of the CAIS, the current practice of building CAIS is performed piece by piece and on demand and on an ad-hoc basis. There are companies outsource their CAIS projects but end up with massive maintenance work to deal with. Other companies buy the off-shelf CAIS software packages but cannot manage to tailor the codes to fit their requirements and have to limit the use of the software packages. These approaches mean waste and failure in the long run [2][5][9]. None of these are satisfactory nor adequate to corporate users.

In this study, we propose a hybrid solution to the above issues and present a simple, standard, and scaleable CAIS design. We develop a hybrid CAIS design which incorporates the job order cost system, the process cost system, and the standard cost system. The design is formulated with the process model and the data model of CAIS. We include the different cost bases, the different cost flows, and the different cost procedures. The design is based on the basic concepts of the bottom-up analysis and top-down design to capture and extract modules. It is developed from the fundamentals of functionality, classification, and commonality. The hybrid CAIS design is module-based. The core of the design is a three-level hierarchy of modules. These hierarchical modules can be classified into (1) the cost accumulation modules, (2) the cost determination modules, and (3) the cost bookkeeping modules. These modules tackle the issues imposed by the different cost bases, the different cost factors, the different cost flows, and the different cost procedures which are mandatory and proprietary to the job order cost system, the process cost system, and the standard cost system. We classify them and organize them into a hybrid design hierarchy. The interoperability and interchangeability between these methods is enabled through the integral database schema. The integrated data model is designed based on the relational model (RM) concepts. Data sharing, integration, and independence are the key principle. We analyze and articulate data items and data properties from each hierarchical module. We apply the entity relationship diagram (ERD) to do the logical database design for the modules. We organize and structure these entities and relationships to develop the CAIS database schema. Figure 1 and Figure 2 show the CAIS design concepts.

This paper is organized into five sections. The first section introduces this study. The second section presents the hybrid design, describes the module hierarchy, and details the process model. The third section defines and delineate the CAIS database schema as a data model. The fourth section describes an implementation of the hybrid design as a system prototype. The fifth section concludes this paper with a brief summary and the future research efforts.

2. CAIS Hybrid Design

2.1 A Three-Level Module Hierarchy

The CAIS hybrid design is modelled with a three-level module hierarchy. The hierarchical design is composed with the process design and the data design. The

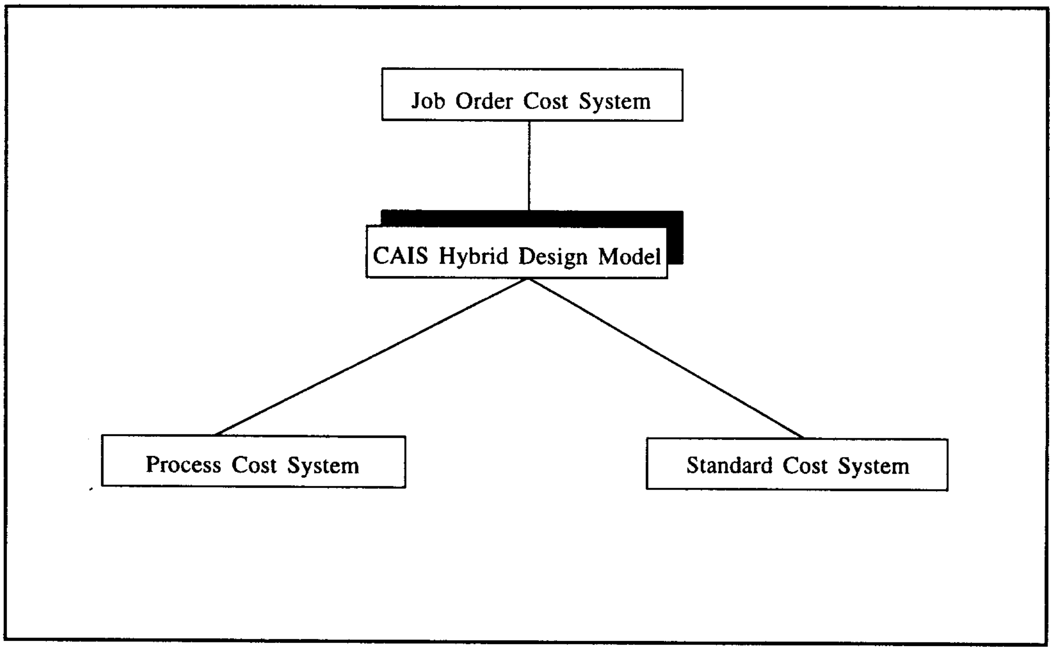

Figure 1: Three Cost Systems in CAIS

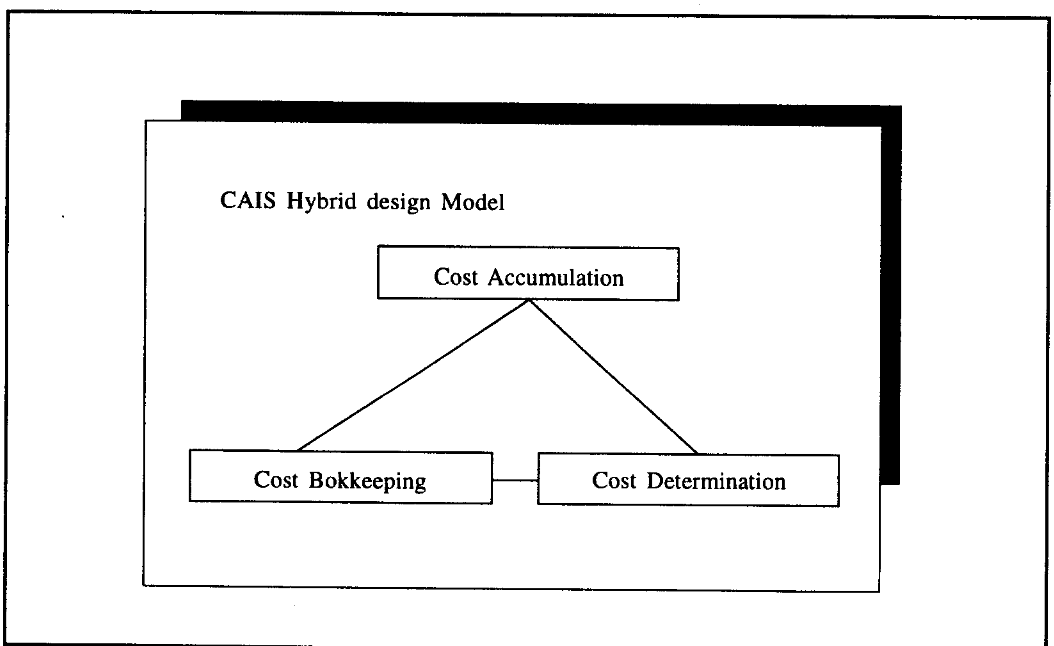

Figure 2: CAIS Hybrid Design Model

concepts of the bottom-up analysis, the top-down design, the fundamentals of modularity, functionality, and commonality form the three-level hierarchy [2][6][10]. We collect and classify the modules into three layers. The first layer is the subsystem layer which heads the entire CAIS development. The second layer is the manager layer which maps out the subsystem functions. The third layer is the module layer which executes the manager procedures as shown in Figure 3.

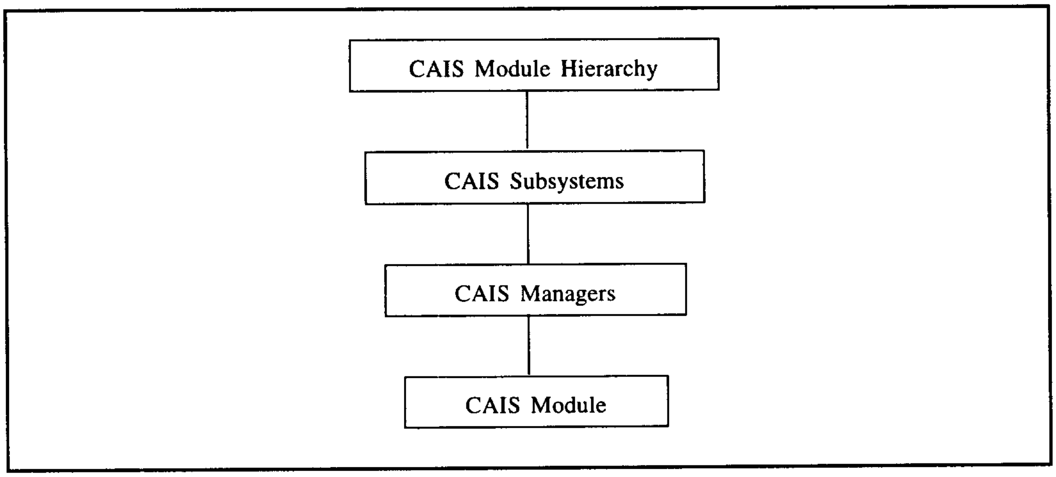

Figure 3: CAIS Module Mierarchy

2.2 Level-One Subsystem Layer

In the hybrid design, we model the first level to be consisted of a set of five CAIS subsystems heading a variety of manager functions. These subsystems are (1) the transaction managers of cost systems to manage the different transaction entries and records, (2) the data managers of cost systems to handle the different data requirements and retrievals, (3) the transformation managers of cost systems to handle the transitions and conversions between different cost systems, (4) the report managers of cost systems to manage the output requirements, and (5) the system utility managers to deal with the system administrative tasks as shown in Figure 4.

2.3 Level-Two Manager Layer

The second level of the module hierarchy consist of a multi-set of 12 functional managers in accordance with the different cost systems and functions. They are (1)

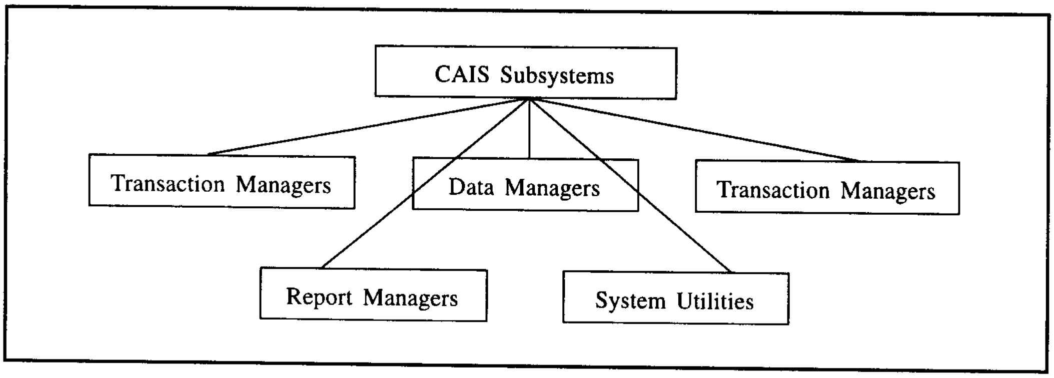

Figure 4: Level-One Subsystem Layer

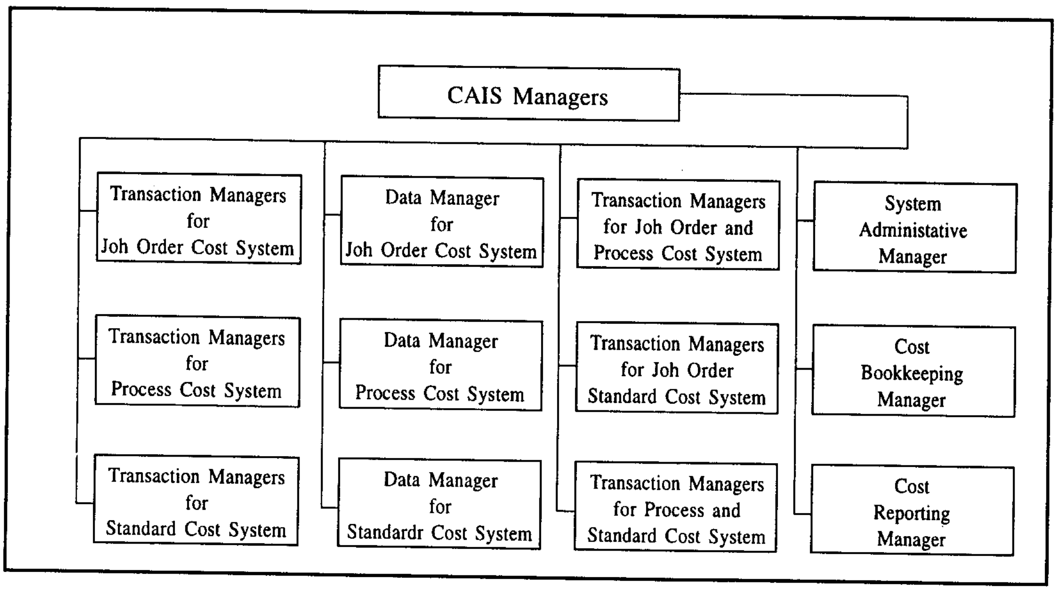

Figure 5: Level-Two Manager Layer

the transaction manager of job order cost system, (2) the data manager of job order cost system, (3) the transaction manager of process cost system, (4) the data manager of process cost system, (5) the transaction manager of the standard cost system, (6) the data manager of the standard cost system, (7) the transformation manager between job order and process cost systems, (8) the transformation manager between job order and standard cost systems, (9) the transformation manager between process and standard cost systems, (10) the cost bookkeeping manager, (11) the cost reporting manager, and (12) the administrative manager as shown in Figure 5.

More, at the second-level, under the transaction manager, the transaction entries and records are divided into 12 transaction types which are to be processed, updated, and stored. These transaction types consist of (1) the cost methods selection, (2) the cost factors determination, (3) the cost allocation base, (4) the order entry, (5) the material taking data entry, (6) the time card data entry, (7) the factory overhead data entry, (8) the standard card data entry, (9) the variance analysis, (10) the work-in-process goods data entry, (11) the finished goods data entry, and (12) the cost accumulation base. And furthermore, at the same level, under the cost bookkeeping manager, CAIS is to perform (1) the slip processing, (2) the journal processing, (3) the general ledger processing, and (4) the subsidiary ledger processing.

At the cost reporting manager, CAIS is to produce eight general and standard reports such as (1) the job order cost sheet required by the job order cost system, (2) the production cost sheet required by the process cost system, (3) the standard cost sheet required by the standard cost system, (4) the variance analysis report required by the standard cost system, (5) the material usage analysis report common to different cost systems, (6) the work-in-process goods analysis report common to different cost systems, (7) the finished goods analysis report common to different cost systems, and (8) the manufacturing cost analysis report.

The transformation manager manages the transition and conversion between different cost systems. The transition and conversion include the data and transaction changes. We provide a mechanism to change data format and data contents from one cost method to another. We change the cost accumulation, allocation, and computing from one cost method to another. We support the retrieval of data from one cost method to be used and stored into another cost method. We utilize the integral data model to facilitate the interoperability and interchangeability of cost accumulation, cost determination, and cost bookkeeping. In the meantime, the integral data model supports the inputs and outputs flowing and binding among the transaction managers, the data managers, the bookkeeping managers, the transformation managers, and the reporting managers.

2.4 Level-Three Module Layer

The third level expands and explodes the second level of modules to give five functional modules which are common uses to the second level managers. These common

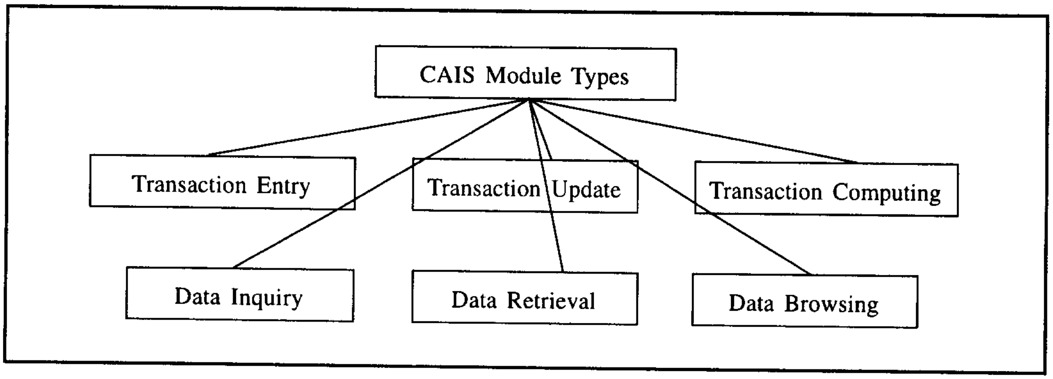

Figure 6: Level-Three Module Layer

modules include (1) the transaction entry module to enter different cost system transactions, (2) the transaction update module to retrieve and change the data of different cost systems, (3) the transaction computing module to get cost accumulation and allocation data and compute the cost sheets for different cost systems, (4) the data inquiry module to inquire the CAIS database, (5) the data retrieval module to get data records with predicates, and (6) the data browsing module to browse through the CAIS tables as shown in Figure 6.

3. CAIS Database Schema

The CAIS database schema is an integrated model of the data definitions for the hybrid CAIS design. The data model incorporates the data requirements of each module in the hierarchy. We use the entity relationship diagram to portray these entities in the CAIS and show the links between different entities. These entities and relationships are divided into three groups of data files, i.e. the cost transactions, the cost computation, and the cost bookkeeping.

The first group of files relates to the cost transactions. These files are "customer", "vendor", "employee", "transform", "take_material", "time_card", "factory_overhead", "standard_card", "inventory". The second group of files regards the cost computation. These files are "cost_sheet", "production_sheet", "work_in_process", "finished_goods", "take_inventory", "variance". The third group of files concerns the cost bookkeeping. These files are "account", "subaccount", "ledger", "subledger", "slip", "journal". The entity relationship diagram of the 22 tables is shown in Figure 7.

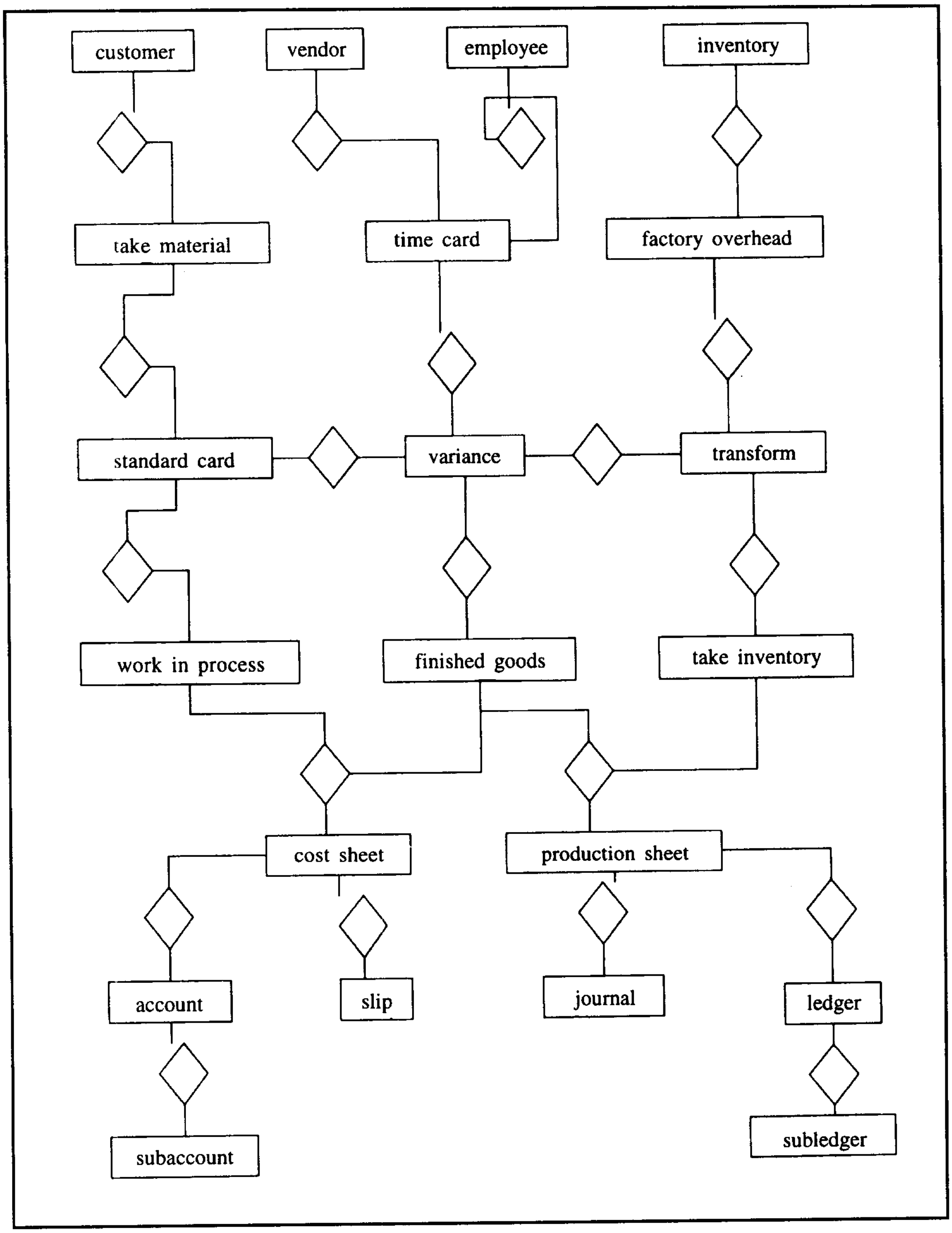

Figure 7 ERD of CAIS Database Schema

We will describe each entity as a relational table file with attributes in the following sections. However, in order to minimize the redundant data definitions with the above, we will introduce and define the tables as those related to cost bookkeeping, those related to cost masters, and those related to cost determination.

3.1 Cost Bookkeepting Tables

There are 21 relational tables which are created to record the data definitions. The first set of seven tables relates to the cost bookkeeping. The first table is the "account" table which has fields of account_id, account_name. The second table is the "subaccount" table which has fields of subaccount_id, subaccount_name. The third table is the "slip1" table which has fields of slip_no, slip_date. The fourth table is the "slip2" table which has fields of slip_no, slip_date, account_no, amount. The fifth table is the "journal" table which has fields of journal_no, serial_no, journal_date, account_no, subaccount_no, debit_amount, credit_amount, balance. The sixth table is the "ledger" table which has fields of ledger_no, serial_no, journal_no, account_no, subaccount_no, debit_amount, credit_amount, balance. The seventh table is the "subledger" table which has fields of subledger_no, ledger_no, serial_no, subaccount_no, debit_amount, credit_amount, balance.

1. account (account_id, account_name)
2. subaccount (subaccount_id, subaccount_name)
3. slilp1 (slip_no, slip_date)
4. slip2 (slip_no, slip_date, account_no, amount)
5. journal (journal_no, serial_no, journal_date, account_no, subaccount_no, debit_amount, credit_amount, balance)
6. ledger (ledger_no, serial_no, journal_no, account_no, subaccount_no, debit_amount, credit_amount, balance)
7. subledger (subledger_no, ledger_no, serial_no, subaccount_no, debit_amount, credit_amount, balance)

3.2 Cost Master Tables

The second set of four tables associate with the master data files. The eighth table is the "customer" table which has fields of customer_no, customer_name, address, phone, contact, credit_line, balance. The ninth table is the "vendor" table which has fields of vendor_no, vendor_name, address, phone, contact, credit_line, balance. The tenth table is the "employee" table which has fields of employee_no, employee_name, date_of_birth, address, phone, hire_date, pay_rate, over_time_rate. The eleventh table is the "transform" table which has fields of transform_no, transform_date, transform_from, transform_to, amount, quantity.

8. customer (customer_no, customer_name, address, phone, contact, credit_line, balance)
9. vendor (vendor_no, vendor_name, address, phone, contact, credit_line, balance)
10. employee (employee_no, employee_name, date_of_birth, address, phone, hire_date, pay_rate, over_time_rate)
11. transform (transform_no, transform_date, transform_from, transform_to, amount, quantity)

3.3 Cost Determination Tables

The third set of eleven tables regards cost accumulation and calculation. The 12th table is the "time_card" table which has fields of employee_no, order_no, process_no, date, start_time, end_time, normal_hours, over_hours, direct_labor, indirect_labor. The 13th table is the "take_material" table which has fields of take_material_no, date, order_no, process_no, material_no, quantity, direct_material, indirect_material. The 14th table is the "inventory" table which has fields of inventory_no, inventory_name, location, date, cost, quantity, balance. The 15th table is the "take_inventory" table which has the fields of take_no, take_date, inventory_qoh, inventory_taken, take_person. The 16th table is the "factory_overhead" table which has the fields of factory_overhead_no, factory_overhead_name, date, amount, order_no, process_no. The 17th table is the "standard_card" table which has fields of standard_no, order_no, process_no, date, standard_capacity, standard_direct_labor, standard_direct_material, standard_factory_overhead. The 18th table is the "variance" table which has fields of variance_no, date, standard_no, order_no, process_no, variance_name, amount. The 19th table is the "work_in_process" table which has fields of work_in_process_no, work_in_process_name, date, quantity, order_no, process_no. The 20th table is the "finished_goods" table which has fields of finished_goods_no, finished_goods_name, date, quantity, order_no, process_no. The 21st table is the "cost_sheet" table which has fields of cost_sheet_no, order_no, date, direct_material_amount, direct_labor_amount, factory_overhead_amount, unit_cost, unit_price, finished_goods, sales. The 22nd table is the "production_sheet" table which has fields of production_no, process_no, prior_process_no, next_process_no, direct_material_equivalent, direct_labor_equivalent, factory_overhead_equivalent, prior_equivalent, beginning_equivalent, unit_direct_material_cost, unit_direct_labor_cost, unit_factory_overhead_cost, unit_prior_cost, unit_beginning_cost)

12. time_card (employee_no, order_no, process_no, date, start_time, end_time, normal_hours, over_hours, direct_labor, indirect_labor)
13. take_material (take_material_no, date, order_no, process_no, material_no, quantity, direct_material, indirect_material)
14. inventory (inventory_no, inventory_name, location, date, cost, quantity, balance)
15. take_inventory (take_no, take_date, inventory_qoh, inventory_taken, take_person)
16. factory_overhead (factory_overhead_no, factory_overhead_name, date, amount, order_no, process_no)
17. standard_card (standard_no, order_no, process_no, date, standard_capacity, standard_direct_labor, standard_direct_material, standard_factory_overhead)
18. variance (variance_no, date, standard_no, order_no, process_no, variance_name, amount)
19. work_in_process (work_in_process_no, work_in_process_name, date, quantity, order_no, process_no)
20. finished_goods (finished_goods_no, finished_goods_name, date, quantity, order_no, process_no)
21. cost_sheet (cost_sheet_no, order_no, date, direct_material_amount, direct_labor_amount, factory_overhead_amount, unit_cost, unit_price, finished_goods, sales)
22. production_sheet (production_no, process_no, prior_process_no, next_process_no, direct_material_equivalent, direct_labor_equivalent, factory_overhead_equivalent, prior_equivalent, beginning_equivalent, unit_direct_material_cost, unit_direct_labor_cost, unit_factory_overhead_cost, unit_prior_cost, unit_beginning_cost)

4. A CAIS Implementation

We build a CAIS system prototype based on the proposed data model and process model, using the Microsoft Visual Basic 5.0 and Accesson Windows NT 4.0. The

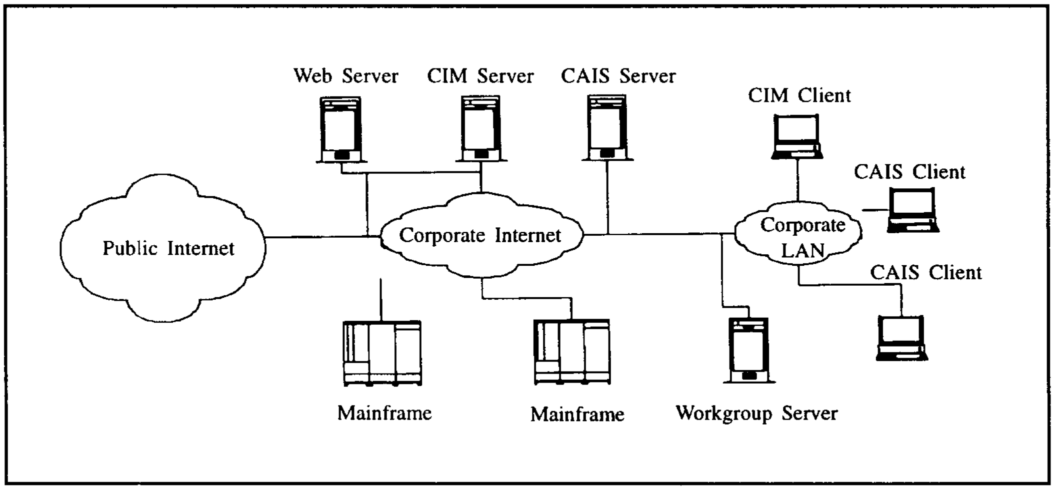

Figure 8 A CAIS System Prototype and Environment

prototype consists of 5 subsystems, 12 managers, and 18 modules. The prototype database has 22 data files. It implements the hybrid design of CAIS as shown in Figure 8.

This prototype shows the simplicity, standardizability, and scalability of CAIS. The design is simple. There are only three levels of process design to deal with. All modules have been analyzed, classified, and standarized. Common services are identified and defined. Modules at the lowest level are coded in the fourth generation programming language (4GL) to give the system portability. Modules which at the lower level evolve over time with more and more common uses can migrate upwardly to the upper level. The reverse is also true for downward migration. When more and new requirements are discovered, we add new modules and upgrade the existing modules based on the principle of generality and specialization to keep the simple and standard design. Further, the integral database schema which is based on the relational model gives the design the property of scalability. It is easy and straight forward to create or drop tables, and to insert or delete records.

This prototype shows the advantages of interoperability and interchangeability. Through the central database, the transformation between different cost methods is achieved. Through the central database and the hierarchic modules, the integration between cost accumulation, determination, and bookkeeping is accomplished. These desirable properties are illustrated in Table 1 as the functional properties and Table 2 as the system properties.

	bottom-up analysis	top-down design	functionality	commonality	modularity
interoperability	X	X	X	X	X
interoperability	X	X	X	X	X

Table 1: Functional Properties of CAIS

	bottom-up analysis	top-down design	functionality	commonality	modularity
simplicity	X	X	X	X	X
standarization	X	X	X	X	X
scalability	X	X	X	X	X

Table 2: System Properties of CAIS

5. Summary

We have presented in this paper a hybrid design of computer-based cost accounting information systems. We have described the detailed design of the process model and the data model. We have accommodated the job order cost system, the process cost system, and the standard cost system in the design. The hybrid design has an integration and transformation center to change between these three different cost systems. We apply the concepts of bottom-up analysis and top-down design to capture and define modules. Common service modules are formed and subjected to at the higher level. Cost bases, cost flows, cost procedures are structured at the management level.

This design has presented a solution to the proprietary issue of CAIS. We create a hybrid CAIS design which becomes the interoperable and interchangeable development template. The template can be portable and scalable across different platforms. From the system prototype, we show that it is viable and doable to build a simple and standard CAIS software package. The simplicity and generality can facilitate the automation of a total manufacturing management information system (MMIS). Our future research efforts will focus on the work-together between the CAIS and CIM, and will focus on the managerial and strategic uses of CAIS to measure the effects of this design on the corporate performance and productivity.

References

[1] Atkinson, A. A., Banker, R. D., Kaplan, R. S., and Young, S. M., Management Accounting, Prentice Hall Inc., 1995.
[2] Burch, J. G., Systems Analysis, Design, and Implementation, Boyd and Fraser Inc., 1992.
[3] Cooper, R., and Kaplan, R. S., The Design of Cost Management Systems, Prentice Hall Inc., 1991.
[4] Gelinas, U. J. and Oram, A. E., Accounting Information Systems, 3rd Ed., South Western Inc., 1996.
[5] Hilton, R. W., Managerial Accounting, McGraw Hill Inc., 1994.
[6] Hoffer, J. A., George, J. F., and Valacich, J. S., Modern Systems Analysis and Design, Benjamin/Cummings Inc., 1996.
[7] Horngren, C. T., Foster, G., and Datar, S., Cost Accounting, Prentice Hall Inc., 1994.
[8] Kaplan, R. S., and Atkinson, A., Advanced Management Accounting, Prentice Hall Inc., 1989.
[9] Kendall, K. E., and Kendall, J. E., Systems Analysis and Design, Prentice Hall Inc., 1992.
[10] Whitten, J. L., Bentley, L. D., and Barlow, V. M., Systems Analysis and Design Methods, Irwin Inc., 1994.