PN ABN 980
ISN 82117

EVALUATION OF SELECTED MIS PROGRAMS OF THE MINISTRY OF HEALTH, REPUBLIC OF INDONESIA

May 5, 1991

Report No. 37

PREPARED BY : Michael L. Gonsalves

International Science and Technology Institute, Inc.
1129 Twentieth Street, NW ▪ Suite 800 ▪ Washington, DC 20036
Telephone: 202-785-0831 ▪ Fax: 202-223-3865 ▪ Telex: 272785 ISTI UR

PN-ABN-980

LIST OF ACKRONYMS

IS	Information Systems
ISO	International Standards Organization
JPKM	Jaminan Pemiliharaan Kesehatan Masyarakat (Indonesian Managed Health Care Program)
LAN	Local Area Network
MIS	Management Information Systems
SSN	Social Security Number

INTRODUCTION

The motto of the Republic of Indonesia is "Unity Through Diversity." As I write this report, I am struck by the idea that this political ideal can also serve as the ultimate goal of information systems and development among the health care financing projects of the Ministry of Health. Just as the Republic must unify its diverse people in order to succeed as a nation, the Ministry of Health must build and unify diverse information systems in order to conduct successful health care programs.

Objectives

The objectives of this consultancy are as follows:

1. Discuss methods and considerations in systems design for MIS.

2. Assess project's information needs.

3. Evaluate current developments and operations, in particular:

   a. Adequacy of hardware systems,
   b. Adequacy of software systems, and
   c. Plans to meet future demands.

4. Review proposed information systems.

5. Identify system design problems.

6. Identify other problems.

7. Recommend methods to resolve the problems.

8. Make recommendations for implementation of LAN based database functions to support project planning, management, and accounting.

Approach

In evaluating these MIS systems and formulating my recommendations, I have set the following guidelines for myself:

1. Sensitivity to Local Culture. I recognize that my view of things is based on the American culture and the ways I have learned to do things in the United States and in Hawaii. But the American way may not be the best way for an Indonesian MIS manager. In my approach to this consultancy, I have tried to minimize cultural bias.

2. Appropriate Technology. Before making a recommendation, I ask myself, "Is this appropriate for Indonesia?" Sound solutions will take into account the differences in culture, economics, and technological infrastructure between Indonesia and the USA. Any methods borrowed from the USA should be adapted to the local culture and business climate.

3. Self-Sufficiency and Self-Reliance. My aim is to enable the client MIS managers and staff to handle their tasks without increasing their reliance on outside resources.

4. Simplicity. Other things being equal, a simple solution is preferred over a complex one.

5. Practicality. I want to avoid grand schemes which have little hope of being realized. My recommendations should be practical and feasible.

GENERAL COMMENTS

The terms, "Information Systems" (IS) and "Management Information Systems" (MIS) are used often and in various ways, sometimes interchangeably. Let me begin by a brief discussion of these and what I mean by them in this report.

Information Systems

In general, information systems (IS) consist of people, procedures and equipment designed to collect, process and present information. They may be manual or automated. A manual information system employs file clerks, typists, secretaries, and other administrative personnel. An automated information system uses computer hardware, software and a variety of specialized personnel. These include data entry clerks, computer operators, programmers, systems analysts, systems designers, technicians, systems engineers, and various technical managers. This report deals only with automated, that is, computer-based, information systems.

Information systems are sets of related records and procedures which are managed and performed according to logically consistent rules and which produce information as their output. The elements of an information system are hardware, system software, application software, databases, procedures, and personnel. The collection of applications software is sometimes called the application portfolio. Those who benefit from information systems, the users, can be anyone in the organization who needs information in order to make a decision, take action, or monitor a situation. The value of an information system is in its ability to output useful information.

Practically every organization has some type of information system, although it may not be identified as such. Some information systems are very informal; others are very formal. Some function in a haphazard, inconsistent and inefficient manner. Even those which employ computers can have these undesirable characteristics.

Usually, an information system is structured along similar lines to the organization which it serves. Just as the organization is divided into functional areas, so is the information system. For example, some common subsystems encompass transaction processing, operational control, management control, and strategic planning.

It is instructive to look at the leading causes of IS failure. First and foremost of these is incorrectly or incompletely defined user requirements. If IS professionals do not understand the problem, it is very unlikely that they will produce a good solution. Another leading cause is rigidity. Both the organization and technology are constantly changing. Overly rigid procedures and rules can hamper adaptability and innovation.

Unnecessary complexity can be very detrimental. Large, complicated systems consume enormous amounts of time and resources to build. Software projects in general are very difficult to be estimated,

both as to cost and schedule. Large, complex projects are even more difficult. When it is necessary to build a complex solution because that is the only way to meet the requirements, the project must be conducted in a very systematic way. Project management is extremely important.

Ignoring human and social factors can result in disaster. A system that is logically correct and efficient from an engineer's point of view is practically useless, if it is too difficult to understand or use.
IS professionals must recognize the importance of both the programs and the database. It is easy to become caught up in automating current business procedures and to look at the problem from a strictly procedural point of view. This can result in a poorly designed, inconsistent, redundant, chaotic and generally inadequate database. It would be miraculous, if the programs written to use that database were clear, understandable, efficient and elegant. A good database design will result in clear, understandable programs. Even the best programmers can be doomed if saddled by a poor database design.

Planning is a key element of any IS operation. It can be done at several different levels, including strategic, tactical, and implementation. One extremely important aspect of planning in information systems is capacity planning. IS management must periodically review the capacities of the various information systems and compare
these with anticipated future needs. The areas in which this is especially critical include mass storage, both on-line and off-line, availability of ports on multi-user systems, and processor through-put. Failure to conduct IS planning will result in failure.

There are many kinds of information systems: financial, operational, manufacturing, marketing, cost management and so on. For example, the Blue Cross/Blue Shield Plan, by whom I was formally employed, use the following subsystems:

1. Finance
2. Funds Accounting and Management
3. Enrollment
4. Statistics
5. Underwriting
6. Member Service
7. Marketing
8. Cost Management/Containment

Management Information Systems (MIS) are a particular kind of information system which support the overall management of the organization.

MIS Overview

Management information systems are a specialized breed of IS designed to support executives in the overall management of the enterprise. They tend to be comprehensive in scope, addressing all aspects of the organization's business, rather than focused on a particular activity.

MIS exist to support management activities and functions - operational control, management control, strategic planning. They encompass information storage and retrieval, command and control, data base management and decision support subsystems. A decision support system supports the various stages of the decision-making process: intelligence (search and discovery), decision design (generating alternatives), and choice.

Data resources are critical to the effectiveness of an MIS. If data are not readily available, the MIS can be crippled. Typically, the data used by an MIS are fed from other business systems, such as finance

and accounting, statistics, underwriting, and membership. In order to be timely and effective, an MIS must be linked to the other information systems of the enterprise.

The first efforts to establish an MIS often involve producing management control reports. Reporting can be either routine or by exception. At some point, there will be a desire to make ad hoc queries. An organization's information systems typically evolve through predictable stages:

1. Isolated applications spring up to accomplish specific tasks, usually the automation of clerical tasks;

2. Data-sharing between separate applications;

3. Integration of separate applications which reside in the same functional area into subsystems;

4. Data-sharing between subsystems, including a nascent MIS; and

5. Integration of separate subsystems into an enterprise-wide information system, including a "full-blown" MIS.

In an organization with a well-developed IS, there may be an enterprise-wide information system which encompasses all the other information systems, including MIS, as subsystems. An enterprise information system integrates all IS functions into a single whole.

Example: Blue Cross/Blue Shield

Systems Development

A great deal already has been written on how to develop information systems. This report is not a text on this subject, but I do wish to touch upon certain key points.

There is nearly universal agreement that any systems development project should be divided into discrete phases. This is often referred to as "The Phased Approach."

The first phase should be an **assessment of needs.** In this phase, one seeks to define the project objectives. It would be wise to publish a document describing the project's objectives and obtaining a consensus of all the parties involved before moving on to other tasks. Once the objectives are understood, the scope should be defined. One must make clear demarcations to separate this project from other projects or other needs that will not be addressed in the project. The final product of the assessment of needs should be a document specifying the features required to meet the project objectives.

The second phase is **analysis.** Here we strive for precise, specific definitions. The generalities of the assessment of needs must now be distilled into concrete, specific language. The analysis phase encompasses both processes and data — that is, we analyze from two different and complementary points of view. From the process view, we look at the procedures and steps that must be taken to accomplish the system's objectives. It may help to think of the procedures and responses to events, to ask "To what events must the system respond?" From the data view, we look for the data elements that must exist in order to accomplish the objectives.

The third phase is **design.** In the design phase, we construct procedures and databases which conform to the preceding analysis. The database design is particularly important and should be tackled first. With a correct database design in place, procedure design becomes easy. Without a good database design, procedures often become muddled in complexity that often misses the mark.

What should be considered in database design? First one must determine the type of database to be employed. Our options include flat file, hierarchical, network and relational. I strongly recommend the relational type.

The technical definition of a relational database involves the predicate calculus and higher math than most may wish to deal with. For a simple definition, I suggest the following:

> A database in which the data are organized into tables, each record (or row) to be addressed by one or more keys. The tables are organized so that each data element is represented only once. Records are cross-referenced by links (or relations) where one record in a particular table contains the key of another record in a different table.

The advantages of the relational model are the elimination of redundancies and inconsistencies, and the ease of ad hoc reporting.

In designing the database, these simple steps should be followed: It is important to determine what is needed and how it will be used. The data should be categorized into groups. Determine how the entities relate to each other. List the important facts about each entity. The end product of these initial efforts is a "data dictionary." Next, the data elements must be organized in a logical manner into tables. Keys must be selected for each table. Then the key fields should be used to establish links between tables.

There are many pitfalls to be avoided in database design. Avoiding them in the early design stages, instead of catching them later, during testing, for example, will save enormous amounts of time and money. Here are the three most common:

1. Missing data — Retro-fitting the database and programs to account for a missing data element is much more difficult than designing it in from the beginning. One should be as thorough as possible in the initial design.

2. Redundant data — Duplication of data in several different places is wasteful and inefficient, creating unnecessary overhead. It can also lead to inconsistencies.

3. Inconsistent data — Inconsistencies often result from redundancies. Especially if the same data are stored in two different formats.

The fourth phase is **prototyping**. Here we build simple, scaled-down versions of the system we have designed. The idea is to give both the IS personnel and the users an opportunity to "walk-through" the application and evaluate it before additional resources are expended.

The fifth phase is the actual walk-through or **evaluation.**

The sixth phase is the **refinement** of the designs.

The seventh phase, one **builds** the full-scale system.

The eighth phase is **testing.**

Conversion is the ninth phase.

Training is the tenth phase.

Implementation is the eleventh phase. If possible, this should begin by running the new system in parallel with the old. Once both IS staff and users feel confident that the new system is performing well, a cut-over date can be set. From that point on, the system is in **production.**

The final phase is **maintenance.** This phase continues on for the life of the system, until it is discontinued.

From the forgoing discussion, it would appear that systems development should proceed in a linear manner, by the numbers, from one step to another. This may be the ideal, with perfect foreknowledge and every detail sliding into place. As a practical matter, this is almost never the case. Back-tracking and iteration are the norm and should be expected. If there is no backtracking, no reiteration of previously accomplished tasks, management should be concerned and investigate.

User involvement at each stage is very important. Before a step is considered completed, both the IS staff and the user representatives should "walk-through" that stage of the application (as if it were a house they were going to buy). This periodic evaluation will protect the organization from discovering major problems long after they first occurred.

<u>Project Management</u>

In order to develop information systems efficiently, I strongly recommend the use of formal project management techniques. Any non-trivial information systems project can quickly go over budget and beyond schedule. A formal project management approach will make these outcomes less likely.

Again there are many texts on how to manage projects. I will touch upon the key elements only. Let us first examine the steps involved:

1. Work Breakdown — Define and list the tasks to be done. Break down complex tasks into simpler tasks. Simpler tasks are easier to estimate and manage. Breakdown large tasks into smaller and smaller tasks. Continue the above until all low-level tasks are no longer than two weeks in duration, clearly defined in concrete terms, and easily understood.

2. Dependency Analysis — Determine which tasks cannot be done until other tasks have been completed. Make a list of these. For each task, specify which other tasks are prerequisite. After the list is complete, a network or PERT chart should be produced.

3. Scheduling — Forecast a schedule, given the information at hand. It probably will not be perfect. Expect to revise it frequently.

4. Budgeting — Forecast the project budget. As with the schedule, expect to revise it. Build something in for contingencies. Remember to account for all of the various types of costs you will encounter: fixed, variable, direct, and indirect. Break your expenditures down into logical categories, like personnel, equipment, supplies, and services.

To complement the formal project management techniques, a formal project organization can be adopted. The head of this organization is the project manager. He/she is responsible for the overall success of the project, for completing the project on schedule, within budget, and meeting the project objectives. The project manager may have one or more assistants. They could be responsible for the more technical aspects of project management, producing the PERT chart, for example. Of course, there is a need for administrative and clerical personnel.

Another key position in the organization is the system architect. He/she responsible for the correct technical design and implementation of the system. This position requires great technical competency. Likewise, there may be assistant system architects. All the technical personnel on the project take their direction form the system architect, including systems analysts, systems designers, programmers, technicians, and other technical specialists.

The third key position is "executive sponsor." This person is a high-ranking officer of the organization who will advocate the project to top management and deal with organizational politics. For example, if the cooperation of some other group within the organization is necessary for the project's success, the executive sponsor exerts his or her influence to insure that cooperation. The executive sponsor is not on the project team. He/she is an external guiding force.

Economic Considerations

One of the most important things to learn about systems development is that the time, money and effort spent in the early stages of the development process can result in enormous savings during the later stages. Consider this example: A high-level manager calls a meeting with his operational and systems personnel. He says, "We need a system to automate record keeping." A few questions are asked, notes are taken, and every one leaves the meeting thinking that they understand what needs to be done and how

to proceed. The systems people proceed through a quick analysis and design phase, then hurry on to code. The boss has set a tight schedule, and they want to complete the project on time. Several months pass by, and the systems people are ready to demonstrate their application. The high-level manager is in attendance, along with the operational personnel. Five minutes into the demonstration, the operations supervisor complains that the program does not handle the records in the same manner that they are used to in the present manual system. One of the clerks explains that the terminology used by the systems people is completely different from that used by operations. The boss then asks why they are automating these records, since he had a different kind of recordkeeping in mind.

Does this sound implausible? There are many systems professionals out there who would be ashamed to admit how often something like this does in fact happen. How is the situation corrected? They could try to adapt what they have already done to what the manager and his operations people want. The result is likely to be less than satisfactory. Or they could start the project all over again. Either way, the project will cost more and take longer than it would have if it had been done right, without cutting corners and with proper attention to detail.

The importance of analysis and design cannot be overstated. A problem that is detected in analysis might cost 400,000 rupiah to correct. If it is undetected until the design phase, the cost of correction could be 4,000,000 rupiah; during testing, or, if detected after the system goes into production, the cost of correction could be 4,000,000,000 rupiah. Of course, these costs are hypothetical, but the point is this: The longer it takes to detect an error, the more it will cost and the more time it will take to correct. And the differences we are talking about are not small, they are whole orders of magnitude.

One way to look at this is through diagrams.

(Diagram: Accumulation of development costs over time
Usual Case: The "Bell"
Worst Case: The "Mountain"
Best Case: The "Whale")

The cost of modification varies greatly, depending on when in the cycle the need is detected. For example, suppose one is building an encounter processing system for an HMO (Health Maintenance Organization). During the initial detailing of the requirements, an error is made. The analysts neglect to note that the system must produce a historical record for each encounter which is to be sent to a regulatory government agency. If this error is detected during the initial design of the database and procedures, the cost might be 1,000,000 rupiah. This cost reflects the need to change designs, but does not involve changing program code or existing databases, since they have not been built as yet. If the error is not detected until programming phase is well under way, many more changes are required to correct the problem. Now the cost is, let us say, 50,000,000 rupiah. If the problem were not detected until the testing phase, the cost of correction might be 100,000,000 rupiah. If the problem were detected after the system went into production and a government official asked, "Where is our historical record?," then the cost might be 1,000,000,000 rupiah.

These numbers are, of course, arbitrary examples pulled out of the air. But they illustrate an important point. The more time, effort, and resources invested in the early analysis and design phases, the less likely one is to encounter very expensive and time consuming changes later. In the U.S., we used to see an amusing commercial about automobile repair. The mechanic held up an inexpensive oil filter (or whatever) and said, "You can pay me now (for this cheap little part), or you can pay me (a whole lot more to overhaul your engine) later."

Technical Guidelines

There are certain guidelines which should be followed in the design and implementation of any system.

Compatibility — Systems should be built to conform to existing standards in hardware, software and data communications. These may be official standards published by some recognized authority, such as the International Standards Organization (ISO), or they may be de facto standards due to market dominance. Any decision to depart from or ignore existing standards should be carefully considered and made only with considerable justification.

Multiple Platforms — Systems should be built in such a way that they can be moved, or "ported," to multiple platforms. It would not be wise to build an expensive application using a programming language that was available only on one particular manufacturer's equipment. Ideally, the application could be run on inexpensive PC's initially, then moved to larger systems as needed. For example, an application written in the C programming language could be run on PC's initially, then ported to larger UNIX-based minicomputers, or even mainframes.

Relational Data Model — Current experience indicates that the relational data model offers the best combination of simplicity, elegance, and functionality. There are many good relational products available. And Structured Query Language (SQL), an out-growth of the relational model, is well on it's way to becoming the standard means of manipulating large volumes of data on big systems or distributed systems.

Distributed Databases — The days when a large mainframe could accommodate the computing needs of an entire enterprise are gone. Typically, the information resources of an organization are distributed among a number of machines on a variety of platforms. A key concept is "shared information." Our challenge is to build systems which recognize this fact and are adapted to it.

Networking — Our most powerful tools for dealing with the problem of distributed databases are data communications and computer networks. There are a number of standards, formal and de facto, which should be followed, particularly in the areas of topology, cabling methods, and network operating systems.

Security and Data Protection — All information systems, even those which reside on PC's, should be protected by appropriate security procedures. One aspect of this problem is physical security. The equipment should be protected from theft and vandalism. Another aspect is redundancy. In the event that the equipment is damaged or fails, backup copies of the programs and data should be available, as well as backup systems on which they can be run.

Training — One of the most common mistakes made with training is to regard it as a one-time expense. Training is an on-going activity. Any healthy organization will have an influx of new personnel. Additionally, even the veterans on the staff can benefit from periodic refresher courses. Avoid overly intensive training. There is only so much that can be absorbed at a time. It is wasteful to train people for eight hours at a time if they become "overloaded" after the first three hours. Whatever training program is established, there should be periodic follow up and evaluation of the curriculum, materials, and instructors. A good approach would be to pre-test the participants to access their skills before the class, then post-test to determine the effectiveness of the training.

Before concluding this section, I want to make a few comments about standards. Too often, standards are seen as an all-or-nothing proposition. Either they are enforced strictly, to the point of stultifying innovation and improvement, or they are not enforced at all, resulting in chaos. Standards in IS should not be "cast in stone," nor should they be ignored. They should be enforced judiciously, keeping in mind that there are times when exceptions are justified.

General Recommendations

In the area of operations, I recommend investigating various innovations to reduce the effort in data entry, including character recognition and bar codes. Standardized encounter forms for all programs would also greatly simplify data entry requirements.

The choice of how to derive membership numbers will be very important. There are several options:

1. Social Security Number's (SSN),
2. Other government identification numbers, or
3. Internally generated numbers (specific to each program or the entire Ministry of Health).

If an internally generated membership number is to be used, I strongly recommend the use of check digits to help guard against data entry errors. I think the SSN would be the best choice by far. In the U.S., many health plans decided to create their own membership numbers.

Sometimes the same individual would receive several health plan membership numbers due to having dis-enrolled and the re-enrolled. Often, it is very difficult to determine whether or not two different numbers

belong to the same person. Eventually, it became necessary for health plans to record the SSN as well. The simplest and most efficient solution by far is to use the SSN as the health plan membership number.

The last recommendation I have is the most important, particularly for the IS personnel of the various programs. We all know that there is a trend toward integrated information systems. Initially, the various programs will operate separately, with little or no sharing of data. Perhaps the only thing they will have in common is the need to transmit certain data to the regulating agency, the DepKes SIP. But later, there will be reasons to pool information. For example, several plans may decide to coordinate benefits, or they may enter into a joint venture, or even merge into a single organization. For these reasons, it would be very beneficial for the programs to decide upon common, standard definitions of data elements. This could be accomplished by a professional organization consisting of all the data base administrators of the various projects.

REVIEW OF SPECIFIC PROGRAMS

PHB

Plans are in place on several levels, including strategy, tactics, and implementation. Currently, applications are under development on PC's using dBASE III Plus. After data are collected on the PC's, this is uploaded to a DEC minicomputer. The purpose of this is unclear. COBOL is used on the minicomputer, making data manipulation inconvenient at best. There are several dBASE emulation products available for minicomputers now. Perhaps one of these would run on this platform. In any event, it is unclear what role the minicomputer will play in PHB's information system.

I recommend that data storage be done on a high capacity, PC-based file server, or data base server with high capacity tape backup. The other PC's should be linked to the file server/data base server through a local area network (LAN). I see no reason to continue the practice of uploading data to the minicomputer.

(J) PKTK (ASTEK)

PKTK serves the large industrial work force only. It is administered by a company called ASTEK. Their objectives are to provide services and contain costs. They view MIS as the nerve system of the organization.

ASTEK's IS planning seems well-developed. Application development efforts are currently at the PC level. Membership information is handled by a system based on dBASE III Plus. Statistical reports for management are handled in Lotus 123.

In general, IS development is going well in this program. The development of two different systems using two different tools is, however, inefficient and somewhat redundant. Since the primary application involves data base management, I recommend that all future systems be developed using database tools. Statistical reports can be generated directly from dBASE III Plus or using a statistical tool that will read dBASE III files directly, such as "dBStat."

Dana Sehat

Dana Sehat is a community-based program. Membership consists of informal groups, small industrial employment groups, community groups, cooperatives groups (soya cake workers, meatball workers, vendors, fisherman — originally formed to take advantage of bulk purchasing power) and villages (a cohesive geographical and cultural group). They see themselves as a financial conduit between the community and health-care delivery organizations. Financing is currently subsidized, but these subsidies will decrease in the future. It was unclear to me whether there is a goal to achieve self-sufficiency, a break-even point.

One major obstacle to progress is that Dana Sehat lacks a business plan. As a result, efforts in IS development lack focus and coordination. Various statistical indicators have been identified, such as cost per member per month, cost per hospital service, cost per service day for procedures, and os on. But there were no plans as to how data is to be collected and analyzed.

Data collection is a very important issue, because Dana Sehat serves a very diverse and geographically distributed constituency. Data collection must take place in the field, often in remote locations. In

particular, an encounter form should be designed which is relatively simple and can be filled out quickly and accurately.

Private Sector

St. Carolus is the only functioning program at present. Physicians are salaried. Their methods are similar to that of Kaiser Permanente in the U.S. in that they both underwrite and deliver services. Separate groups handle the collection of dues and the delivery of services. Providers are paid capitation from the financial group. Capitation does not include (cover/account for) indirect costs. St. Carolus is currently building an MIS for their managed care program with 5,000 members.
An HP-3000 is used for in-patient data processing. There are eight PC's, 80286 and 80386-based, with 80 MB hard disks. Data transfer is accomplished by someone carrying a floppy diskette from one machine to the other, "sneaker-net." A variety of well-known PC programs are in use, including RBASE, dBASE III Plus, WordStar, and Lotus 123. The major application is medical records.

The IS program at St. Carolus seems to have made an excellent beginning. I have a few question, however:

> Why are two different relational database products (RBASE and dBASE III Plus) in use? The products are very similar, but their file formats and 4GL programming languages are different. It would be more efficient to consolidate applications development onto a single RDBMS platform.
>
> Do the 80 MB drives provide sufficient capacity? I recommend that they consider installing a Local Area Network (LAN) with a high capacity file server. This would provide secure access of the data to multiple PC users.

PIQ/RS

One of their most important tasks is to implement a system which can derive unit costs. They are moving from general accounting to cost accounting. There is also a need to link with the pharmacy.

The program's objectives are:

1. Diagnose problems
2. Show pilot solutions
3. Implement successful pilot solutions
4. Efficiency
5. Effectiveness
6. Quality
7. Cost Control
   Standards
   a. Service
   b. Physical

IS capabilities vary from hospital to hospital. There is heavy subsidization. Cost recovery is only 17 percent. In the future, they hope to recover 57 percent. Doctors do not like to treat insured patients. It involves too many difficulties.

Their MIS plans are as follows:

1. Phase One — Introduce PC's with billing and financial reports, GL, AR, AO, Pharmacy system, Medical Records, Statistics.

2. Phase Two — Integrate the above into a unified MIS system with internal links to the pharmacy and external links to JPKM; admissions and billing and medical records.

The cost accounting problem is a difficult one. Direct costs are available, but indirect costs are not. There are no plans to collect utilization data and issue reports to JPKM.

SIP

SIP is a regulatory information system. At the time of my visit, it had not been determined which government agency would be charged with this task. Some type of hierarchical organization will be employed. Health care organizations will report to district regulatory offices. These, in turn, will report to a provincial office. The provincial offices will report to a central national office. These reports will be more summarized, containing less
and less detail, the further up the hierarchy they travel. Thus the district offices will need to collect the most detailed data, while the national office will be concerned with summary data only. Paradoxically, the district offices are least equipped to deal with detailed data collection.

The goals of SIP are:

1. Promote better health,
2. Control costs, and
3. Protect health care and monetary investment.

There are a number of issues to resolve. Should the Ministry of Health require that other groups report certain data so as to guarantee that it will be available when the regulatory body can use it? The regulators cannot wait for complaints; they must go out and investigate. Exception reporting will be the primary application.

CONCLUSION

Two of the best recommendations I can make were those given to me by Dr. Mariano of St. Carolus:

1. Start out very simply and
2. Gradually upgrade in the future.

I strongly concur. But how to start and how to upgrade are extremely important questions! The end result should be a collection of well-planned, well-executed, highly integrated information systems supporting health care delivery and finance in Indonesia. This is not likely to come about if the systems are not begun on a firm foundation and if the upgrade path is not enlightened.

The more that can be standardized (without causing stagnation), the better. Standardized encounter form(s) for all government programs would be very advantageous. Standardized definitions of data elements among the various programs would benefit IS development and integration enormously.

Nearly all of the programs had very prudently decided to start with small, PC-based hardware platforms. Conspicuously absent were plans to migrate the PC-based systems to larger platforms with higher storage capacities. I strongly recommend that all programs include capacity planning as an essential aspect of overall IS planning.

Every program I visited could benefit from formal project management techniques. There are several excellent PC applications for project management, such as TIMELINE from SYMANTIC, MICROSOFT PROJECT, and SUPERPROJECT from COMPUTER ASSOCIATES. I strongly recommend that the programs select one and use it.

By far the best possible membership numbering scheme would be to use the already existing social security numbers. They are unique to each person and usually have to be tracked anyway. Creating yet another such identification number would be redundant.

Very careful consideration should be given to the selection of technical platforms for the systems to be built. In particular, manufacturer-specific and proprietary tools should be avoided where possible. Well-established, standardized (formal or de facto) tools, operating systems, network software, programming languages, and so on, are preferred.

We should expect different rates of development for different organizations, or even groups within organizations.
One of the most important decisions to be made is which of the cost curves to embrace. Will you "Climb the mountain," "Ring the bell," or "Ride the whale?"

"Unity Through Diversity." How is that accomplished? Only if there is order within diversity. There can be no unity if there is disorder, irrationality, or chaos. With the proper order, rational, systematic, well-thought-out, there can be progress and achievement.