

# ***THE BEST OF MBA***

## **TABLE OF CONTENTS**

Title & Publisher

Introduction

Some Common Sense about Improving Productivity

Moving IRM from an Art to a Science

The Four Stages of IRM Growth

Corporate Culture

Today's Chief Information Officers: the Untouchables

In Japan - 'Maybe Yes' meand 'No'

The Lady or the Tiger

Back to the Future: Re-Inventing Re-Engineering

An Interview with Milt Bryce

Bryce's Laws on IRM

About MBA

MBA Product Offerings

MBA Service Offerings

IRM Revolution Book

## **ABOUT THIS FILE & DOCUMENT**

This file may be freely copied and distributed; however, the contents (text and graphics) are the property of M. Bryce & Associates, Inc. (MBA). Any reproduction of the contents without the expressed written permission of MBA is strictly prohibited.

"PRIDE" is an acronym for PROfitable Information by DEsign - through phased planning and control. "PRIDE" and "INFORMATION FACTORY" are the registered trademarks of MBA. Copyright (C) M. Bryce & Associates, Inc. (MBA) 1995. All rights reserved.

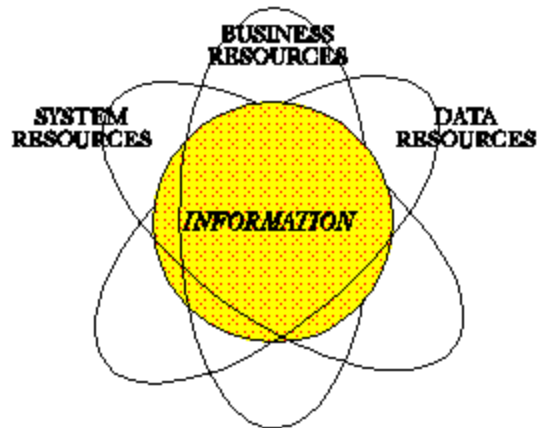
Trademarks:

- \* IBM and OS/2 are the registered trademarks of the International Business Machines Corporation.
- \* Windows is the trademark of Microsoft Corporation.
- \* "PostScript" is a trademark of Adobe Systems Incorporated.

All other trademarks both marked and unmarked belong to their respective companies.

**END**

## ***TITLE & PUBLISHER***



### ***THE BEST OF MBA WRITINGS ON INFORMATION RESOURCE MANAGEMENT (IRM)***

#### **Published by:**

MBA Press

M. BRYCE & ASSOCIATES, INC.  
777 Alderman Road  
Palm Harbor, FL 34683  
United States  
Tel: 813/786-4567  
Fax: 813/786-4765  
BBS: 813/786-4864  
CompuServe: 76235,2364  
E-Mail: TimB1557@aol.com  
IBM Link: DEV2643

Tim Bryce - Editor

Published - June 1995

Since 1971: *"Software for the finest computer - the Mind"*

This file may be freely copied and distributed; however, the contents (text and graphics) are the property of M. Bryce & Associates, Inc. (MBA). Any reproduction of the contents without the expressed written permission of MBA is strictly prohibited. "PRIDE" is an acronym for P<sup>R</sup>ofitable I<sup>N</sup>formation by D<sup>E</sup>sign - through phased planning and control. "PRIDE" and "INFORMATION FACTORY" are the registered trademarks of MBA. Copyright (C) M. Bryce & Associates, Inc. (MBA) 1995. All rights reserved.

Trademarks:

- \* IBM and OS/2 are the registered trademarks of the International Business Machines Corporation.
- \* Windows is the trademark of Microsoft Corporation.
- \* "PostScript" is a trademark of Adobe Systems Incorporated.

All other trademarks both marked and unmarked belong to their respective companies.

END

## ***INTRODUCTION***

Ever since we began MBA in 1971 we have been asked to comment and write on a variety of subjects related to Information Resource Management (IRM). Our articles have appeared in some of the most prominent trade journals in the world. These writings culminated in 1988 with the publication of our book, "THE IRM REVOLUTION: BLUEPRINT FOR THE 21ST CENTURY." Reaction to the book has been gratifying, particularly in Japan where it became a best seller.

In producing this publication, we selected those articles from our collection that garnered the most interest and response from our readers. This is not to suggest that our readers always agreed with us; some of our comments inevitably would create controversy. But bottom-line, we want our readers to think about what they are doing as opposed to blindly following the latest fad or trend.

This collage of articles is primarily aimed at MANAGEMENT. We do not believe another technical journal would provide any assistance in addressing the problems of planning, design, and development. Rather, this publication is oriented to providing insight into the management problems in this area and how to address them. "THE BEST OF MBA," therefore, is intended for an audience of I.S. and executive managers. Application developers will also find it of interest as will end-users.

Although some of the articles were written in the mid-1980's they remain as relevant today as they were when they were first published. We hope you enjoy them.

To make this publication easy to distribute and read, we are providing the articles in three file formats:

- 1. BESTMBA.TXT** - This is a version of the publication in ASCII text format suitable for use with your favorite text editor/word processor. Please note this version does not include the graphics as used in the other two versions.
- 2. BESTMBA.INF** - For use with the standard "View" utility (VIEW.EXE) accompanying IBM's OS/2 (version 2.x or higher).
- 3. BESTMBA.HLP** - For use with the standard "Help" utility (WINHELP.EXE) accompanying Microsoft's Windows (version 3.x or higher).

Both the INF and HLP file formats include graphics and text suitable for on-line viewing. These facilities also provide a convenient means to quickly search through the documentation as desired.

This electronic publication can be obtained from a variety of network providers (file name BESTMBA.ZIP):

- \* America Online
- \* CompuServe
- \* FTP sites
- \* Bulletin Boards (including MBA's own BBS at 813/786-4864)

Address questions and comments to:

M. Bryce & Associates, Inc.  
777 Alderman Road  
Palm Harbor, FL 34683  
United States  
Tel: 813/786-4567  
Fax: 813/786-4765  
BBS: 813/786-4864  
E-Mail: TimB1557@aol.com  
CompuServe: 76235,2364  
IBM Link: DEV2643

## **ABOUT THE AUTHORS**

Unless otherwise noted, the articles represent a collaborative effort between Milt Bryce and Tim Bryce.

### **Milt Bryce**

Mr. Bryce is the President & C.E.O. of MBA. Having begun his career in 1954 with the Univac I, Mr. Bryce was one of the first few programmers in the United States. Over the last four decades, Milt pioneered many concepts and innovations, including: structured design, data resource management, layered systems documentation, methodologies, and data dictionaries, to name but a few. In 1971 he founded MBA and is regarded as the father of the company's "PRIDE" family of products. Mr. Bryce is a frequent lecturer and has spoken extensively to computer and management related societies worldwide. In 1982 he was recognized in Tokyo for improving the productivity of systems development in Japan. Mr. Bryce is a charter member of the Association for Computing Machinery (ACM) and the American Association for Artificial Intelligence. He earned a Bachelor of Arts (BA) degree in Industrial Psychology from the University of Buffalo (now the State University of New York at Buffalo).

### **Tim Bryce**

As the Director of Marketing & Customer Service for MBA, his responsibilities include training and installing "PRIDE" products at customer locations worldwide. In addition, he is the company's chief product architect and was responsible for developing the company's Enterprise Engineering Methodology (EEM), Computer Aided Planning (CAP) aids, automated systems design tool, and the "PRIDE" INFORMATION

FACTORY. In addition to his regular duties, he has been active in the Association for Systems Management (ASM), Founder and Past-President of the Tampa Bay OS/2 Users' Group (TBOUG), and Editor of various newsletters including OS/2 CONNECT, PROS/2, and Management Visions. Mr. Bryce holds a Bachelor of Science degree in Communications from Ohio University, and is a Certified Systems Professional (CSP).

Also mentioned in this publication:

**Kazuya Matsudaira**

As President of PRIDE Japan, Inc. (PJI) in Tokyo, Mr. Matsudaira represents MBA in Japan and the Asian market. As MBA's representative since 1976, he has provided consulting and training services to some of Japan's largest companies. Mr. Matsudaira is frequently quoted in the Japanese trade press and often lectures on IRM related subjects, both at home and abroad. He is a member of the Association for Computing Machinery (ACM), and the Japanese Management Association (JMA). His company is also actively involved with Japan's Ministry of International Trade & Industry (MITI), the Japanese Software Industry Association (JSIA), and the International Standards Organization (ISO). Mr. Matsudaira holds a Master's of Business Administration degree from Keio University.

END

COPYRIGHT (C) MBA 1995

# ***SOME COMMON SENSE ABOUT IMPROVING PRODUCTIVITY***

*June 1, 1995  
by Tim Bryce*

I have been in the systems industry for twenty years now, which to my way of thinking is not a very long time. However, it has offered me the opportunity to witness several technological changes in the area of systems development. In this short period of time I have seen several fads come and go:

- Data Dictionaries
- Structured Programming
- CPA written Methodologies
- Project Management Systems
- Fourth Generation Languages
- Data Flow Diagrams
- CASE tools
- Function Point Analysis
- Information Engineering
- Encyclopedias/Repositories
- Joint Application Development (JAD)
- Rapid Application Development (RAD)
- Total Quality Management (TQM)
- Business Process Re-Engineering (yes, BPR was a "fad")

The current craze is Object Oriented Programming (OOP), the Data Warehouse, and Client/Server Computing. Despite this wide array of tools and techniques I am still amazed that the problems are no different today than when I entered the field in the early 1970's: projects are still late and over budget, documentation is virtually non-existent, data redundancy is still the norm (as opposed to sharing and re-using data in integrated systems), users are unhappy, systems are built that do not adequately serve business needs, even worse, system priorities are not synchronized with business priorities, etc. The development problems of the 1990's are no different than those of the 1970's (or earlier).

Management is bewildered by the return on their technological investment. Even though they have paid millions for the latest computer gizmo, they're at a loss as to why their systems are not running like a fine watch.

Before we start pointing fingers let's consider our perspective on productivity. For years companies have been operating under the premise that the best way to improve productivity is through task efficiency. In other words, if you want to improve the task of welding, you may want to use robotics to perform it; If you want to improve the speed of software development, you buy a better programming tool.

This philosophy is fine for a single task, but what is necessary is the ability to orchestrate all of the tasks in a development project in a concerted manner. Otherwise, tasks are performed at the wrong time in a disjointed, uncooperative manner. As projects fail, the tools are inevitably blamed, purged, new ones purchased, and the vicious circle starts all over again.

There is more to productivity than just efficiency, there is also the concept of effectiveness:

### **PRODUCTIVITY = EFFICIENCY X EFFECTIVENESS**

The differences between efficiency and effectiveness is subtle yet significant. Whereas the former addresses task performance, the latter is concerned with the necessity of the task itself. There is nothing more unproductive than to build something efficiently that should never have been built at all. To illustrate, robotics is an efficient means for performing tasks such as welding, but if a weld is performed at the wrong time or place, it is counterproductive regardless of how efficiently it was performed. The same is true in programming; if a program is written prematurely or outside the boundaries of an overall systems architecture, then it will be counterproductive, regardless of the tool used to produce it.

In a manufacturing facility, the Industrial Engineer is responsible for the organization and layout of the assembly lines and what tools will be deployed throughout them. The engineer focuses on the organization and synchronization of the development tasks (effectiveness). When this is accomplished, the engineer concentrates on the tools and techniques to be used (efficiency).

Unfortunately, most systems development organizations do not have a support function like the Industrial Engineer to layout and monitor the development environment. This is because it is erroneously assumed that everyone understands Who is to perform What tasks, When, Where, and Why (the 5-W's) in a uniform manner. Typically, each project is executed according to the whims and discretion of the individual Project Manager or lead programmer. Uniformity between development projects is avoided, leading to inconsistent and incompatible results between projects. Consequently, the opportunity to share and re-use resources is lost.

In order to move the development of systems from an art to a science in an organization, it is necessary to define and standardize the terminology and establish a consistent development environment. This is complicated by industrial gurus and academics who perpetually try to re-invent the industry by changing the terminology and concepts. As I listen to some of these pundits at trade shows or read their articles, I am reminded of the expression, *"re-arranging the deck chairs on the Titanic."*

A little common sense will go a lot farther in this business than the latest technological fad. Developing enterprise-wide systems need not be cryptic or an arduous task.



What is necessary is someone to define the ground rules for development, and layout the roadmap for how to get from one point to another in an organized manner.

Implementing a defined development environment is perhaps a bigger challenge than introducing a new tool since it represents a change to the corporate culture. Instead of hacking away at program code over and over again, we're now asking people to plan and organize their activities, and impose certain disciplines.

Some people believe that building systems requires an artistic type of person, and that organization, discipline and accountability only serves to stifle the creativity of the individual. This appears to be the mind set at companies such as Microsoft who also consistently manages to slip product delivery dates. However, nothing could be farther from the truth. Even in creative fields such as music, art, and architecture there are certain governing rules and regulations to be observed in development; why should the development of systems and software be any different? Surprisingly, most people are more comfortable working in a defined development environment as opposed to one that is chaotic.

Years ago, when I took my first programming class from IBM, I was put on a team with two other programmers from other companies who had more "on the job" experience than I had. As part of our training, we were given small programming assignments to implement. With each assignment, I would take out my template and draw a block diagram of the processing logic for the program, code it, then compile and debug it. While I was doing this, my two cohorts would simply start coding, compiling and debugging. Whereas it would take them several compiles for them to get it right, I would get it right on either the first or second compile and in much less time. Curious, I asked my teammates why they developed programs the way they did, and why not think the problem through first like I had done. They explained their management didn't want them to waste time planning and flowcharting, that the real work was in coding. This baffled me since I was performing the task quicker in a more disciplined and consistent manner. Later on, I bumped into the IBM rep who signed me up for the class. The rep asked me what I thought about the course. I replied, "I'm learning more about programmers, than about programming."

Twenty years later, I'm still learning about programmers. Even though the development technology has changed, the fundamental philosophy has not. I believe the secret to improved productivity does not lie in efficient programming aids, but rather in managing programmers to perform their jobs more effectively. A little common sense is all that is needed. However, there's only one problem with common sense in this industry; it's not very common.

END

COPYRIGHT (C) MBA 1995



**NOTE:** *This article first appeared in the August 1993 issue of MBA's Management Visions newsletter.*

## ***MOVING IRM FROM AN ART TO A SCIENCE***

*by Milt Bryce, Tim Bryce and Kazuya Matsudaira*

### **Foreword**

Even though western companies have become proficient in developing computer software, building major enterprise-wide systems is an embarrassingly difficult task for most companies to perform. The press is riddled with stories of major systems development projects that have gone awry. Project "runaways" (projects that overrun budgets and schedules) have become common-place in corporate America. Further, patching programs, commonly referred to as "firefighting," has become the normal mode of operation for most IS organizations. Consequently, little progress can be made towards the major development projects required for the business.

In contrast, numerous Japanese companies have successfully been able to design and build major systems on time and within budget. Is it because they used some magical software engineering tool? No. They have simply been able to apply sound management principles to the design and development of their information resources. The results have been overwhelming. Not only have they delivered on time, the Japanese have produced quality systems with high user satisfaction. These systems have been applied to all types of industries; banking, manufacturing, construction, utilities, etc., and they are beginning to have a profound impact on how the Japanese conduct business; e.g., zero defects in production, cost reductions, accelerations in product delivery, cash flow improvements, reduced inventory overhead, improved customer service, etc.

As the Japanese have demonstrated, developing information resources need not be a cryptic art form. Rather, a scientific method can be applied based on common-sense and time honored management techniques. This paper is based on the authors' observations of several Japanese clients. The authors conclude that the methods used by their clients are not necessarily unique to the Japanese culture and can be universally applied to just about any business, East or West.

Since the subject of "Information Resource Management" (IRM) first emerged in the late 1970's, there have been volumes written to describe the concept and underlying theories supporting it. Unfortunately, most explanations have been academic and/or cryptic in nature. Consequently, few companies have adopted a clear IRM policy and capitalized on this powerful management concept.

There are probably as many interpretations of IRM as there are people who tout its virtues. And wherever inconsistencies occur, confusion is sure to follow. There are

those who see IRM as nothing more than records management, others simply regard it as managing data resources, or merely managing information related technologies (e.g., computers, networks, office automation, etc.).

What is needed is a common-sense approach to IRM that executives can readily understand and embrace; one that is not couched in esoteric theories of information management. Whereas some companies have treated IRM as an art form, there have been others who have been able to use common management concepts to turn IRM into a legitimate science.

There are significant differences between an "art" and a "science." An "art" depends on an individual's intuitive instincts about a particular subject. Such intuition is difficult to teach and apply in a consistent manner. An art-form, by definition, implies non-conformity and represents an expression of personal style and taste. In contrast, a "science" is based on proven principles and, as such, can be taught and applied in a uniform manner by many people.

In order for IRM to progress from an art to science, a body of knowledge has to be defined in terms of proven concepts and standard terminology. Unfortunately, this is where the industry has been wallowing for the last fifteen years. Whereas computer vendors and consultants have been arguing over the semantics of IRM, a group of Japanese companies have been able to establish a practical science and put it into practice. Their example reveals that it is not necessary to invent any new theories of management, but rather to re-use existing management principles that have already been proven over time. By doing so, they have been able to successfully move IRM from an art to a science.

## **DEFINING INFORMATION RESOURCE MANAGEMENT (IRM)**

The concept of IRM has been evolving in Japan over the last fifteen years. Today, it is primarily regarded as *the design, development and control of ALL of the resources required to produce information*. This includes three classes of information resources:

**1. BUSINESS RESOURCES** - representing the parts of the business requiring information or involved in the processing of data. These resources include enterprises, business functions, jobs/positions, human/machine resources, skills, business objectives (specifying motivation), and projects (implementation of objectives).

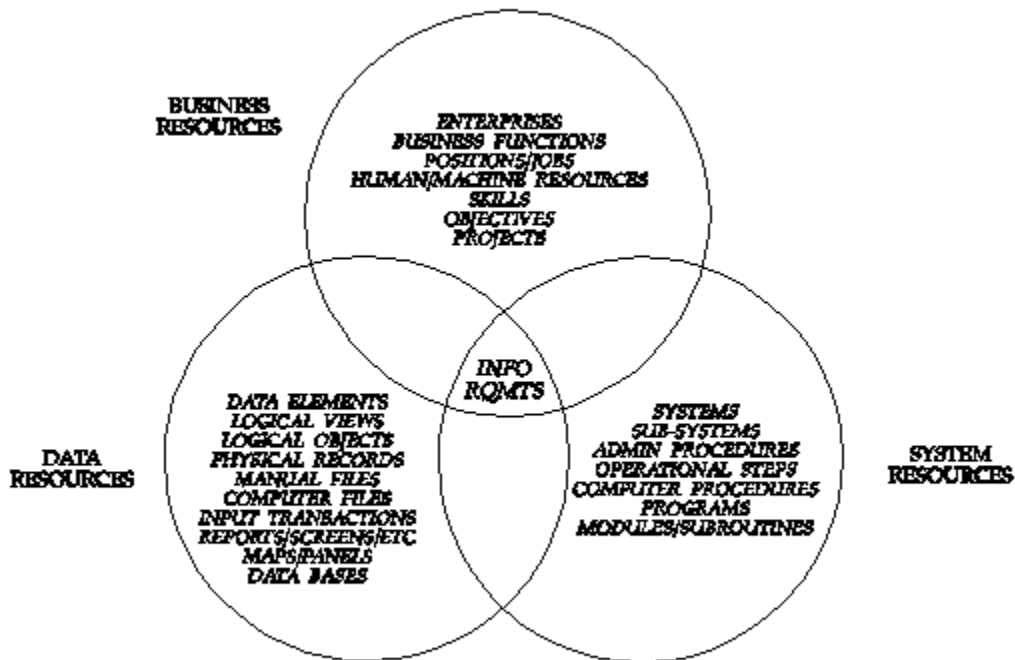
**2. SYSTEM RESOURCES** - representing the necessary processing components. This includes information systems, sub-systems (business processes), administrative procedures (for manual processing and office automation), operational steps, computer procedures, programs, and modules (separate "building blocks" of software).

**3. DATA RESOURCES** - representing the data needed to produce information. Included are data elements, storage records, computer files, manual files, input transactions, screen panels (including windows), print maps, inputs, outputs, logical files

(objects), logical records (views), and data bases.

Control over these resources permits their manipulation to produce different forms of information to serve business.

## **INFORMATION RESOURCES**

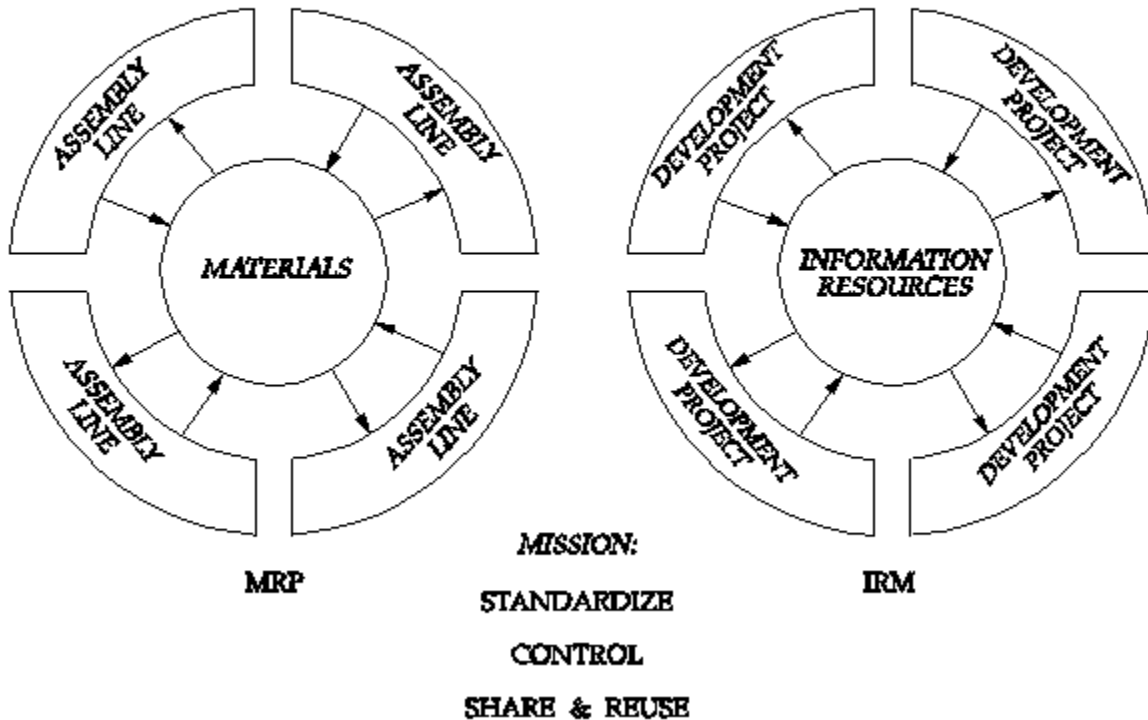


**INTEGRATION IS MADE POSSIBLE THROUGH  
A BILL OF MATERIALS PROCESSOR (BOMP)**

### **THE MRP ANALOGY**

In many ways, the IRM concept is analogous to the discipline of Materials Resource Planning (MRP) as found in manufacturing. MRP is the process where all materials, raw or manufactured, are specified, cataloged and cross-referenced to products, assemblies, sub-assemblies and other parts. Consequently, different products can be easily assembled and inventories effectively controlled. This is precisely the same objective of IRM. But instead of products and parts, IRM is concerned with information and the resources needed to produce it.

# IRM/MRP ANALOGY



The mission of IRM and MRP, therefore, is the same: to standardize, control, share and re-use components needed to produce products. Whereas MRP is geared towards managing the parts needed to produce products, IRM is concerned with managing the resources needed to produce information.

A single product may consist of hundreds or thousands of components. Conversely, a single information system will consist of hundreds or thousands of resources. Trying to manage these resources without some form of standardized and consistent approach will only produce disjointed results and the opportunity to share and re-use resources will be lost.

One of the prime functions of MRP is to coordinate the use of parts on assembly lines. The same is true in IRM where resources are shared and re-used between development projects. Sharing and re-using resources in this manner offers many benefits:

\* Resources can be re-used in other projects, thereby expediting future application development projects. To illustrate, if a data element is properly defined in one application, it should not have to be re-defined in other applications; it should be re-used, which results in time and costs saved. Historically, the trend has been to define data on an application by application basis. This leads to inconsistent definitions and

complicates system maintenance, particularly when dealing with complicated calculations that have been defined differently for each application.






\* Sharing resources promotes systems integration, where multiple systems share data, which improves the integrity of data. A file in one system should be used in other systems, provided it contains the necessary data and can be accessed in the required time frame. This eliminates redundancies and produces consistent facts. For example, it is not uncommon to find such redundancies in corporations. As a result, executives must work with incompatible results (e.g., one executive perceives "gross sales" differently from other executives).

\* Controlling resources provides the ability to effectively study the impact of change of one resource on others, thereby improving change control (implementing changes and corrections). For example, if an information resource such as a module or data element is properly defined and related to other resources, it becomes a relatively simple matter to study the effect of change on all resources. Such analysis is invaluable for planning and implementing changes. Without such analysis, changes are implemented without any forethought. Subsequently, seemingly harmless changes produce catastrophic repercussions and "firefighting" (correcting "bugs") becomes the normal mode of operation.

This IRM/MRP analogy is significant. Not only does it establish a palatable rationale for IRM, but it paves the way for creating a mass production environment for engineering and manufacturing information resources.

There are five basic elements within any mass production environment: Mass Demand, Assembly Lines, Division of Labor, Standardization of Parts, and Precision Tooling. Several Japanese companies have been able to adapt to the MRP analogy and implement mass production environments for building information resources.

# THE FIVE ELEMENTS OF MASS PRODUCTION

	<p><b>MASS DEMAND - THE IMPETUS FOR MASS PRODUCTION. (USER INFORMATION DEMANDS)</b></p>
	<p><b>ASSEMBLY LINE - DEFINING THE PROGRESSION/SYNCHRONIZATION OF WORK.</b></p>
	<p><b>DIVISION OF LABOR - TO BREAK THE PRODUCTION PROCESS INTO SEPARATE TASKS PERFORMED BY SPECIALISTS (HUMAN, MACHINES, BOTH).</b></p>
	<p><b>STANDARDIZATION OF PARTS - FOR INTERCHANGEABILITY AND ASSEMBLY BY UNSKILLED OR SEMI-SKILLED WORKERS.</b></p>
	<p><b>PRECISION TOOLING - FOR MECHANICAL LEVERAGE.</b></p>

## MASS DEMAND

Mass Demand represents the impetus for any type of mass production. Customer demand results in the production of new products and services. The demand for accurate and timely information is no different.

## Kansai Electric Power Company, Inc.

Kansai Electric is a major power utility serving over 11 million customer accounts in the six prefectures of the Kansai district of Japan (Osaka area). In 1964 the company developed its first major computer application, automation of the Customer System. As one of the company's core systems, the Customer System was used to monitor energy consumption and issue customer billings. Like any information system, Kansai's Customer System underwent extensive changes over the next twenty years. Modifications were required to implement changes to tariff structures and computer technology; e.g., the addition of interactive screens, the introduction of new data base management techniques, and migration from one computer platform to another (Burroughs to IBM). After 20 years of sporadic surgery, the Customer System was difficult to manage and maintain. So much so, that by 1987 an executive level decision was made to completely re-design the whole system.



Kansai's information requirements were fueled by the need to service a growing customer base and changes in government regulations. The technical implementation of the system simply could not keep pace with the burgeoning demands for information. Studying the problem in detail, Kansai identified 3,139 information requirements to be implemented by the system. Recognizing that building a system of this magnitude would be no small endeavor, Kansai decided to re-evaluate their development environment prior to undertaking such a mammoth development project.

### **The BEST Project**

By the late 1980's Japanese banking systems seriously lagged behind those in the west. So much so, Japan's Ministry of Finance helped orchestrate a strategic project to leap-frog the west in terms of banking systems. Entitled the BEST Project, the Ministry brought together four of the top trust banks in Japan: Yasuda Trust & Banking, Mitsubishi Trust & Banking, Nippon Trust & Banking, and Chuo Trust & Banking. The intent of the project was to revolutionize Japanese banking systems by creating a cooperative systems environment between the four banks. The premise was to design and share information resources, thereby creating integrated information systems and provide customers with extraordinary service.

Although the premise of the project was simple, the scope was inordinately large. Realizing the project would require the implementation of voluminous information requirements and probably dozens of systems, the consortium decided to appraise their development environment. Compounding the problem was the fact that each company participating in the project used different and conflicting methods for developing systems. Consequently, it was essential to redefine the project's development environment prior to embarking on a project of this magnitude.

In both examples, Kansai Electric and the BEST Project, voluminous information requirements were the impetus for challenging traditional development methods and creating a mass production environment.

### **ASSEMBLY LINES**

The second element of mass production is the Assembly Line defining the progression and synchronization of work. This concept has been successfully applied to everything from automobiles and consumer goods to shipbuilding and construction.

Perhaps the most obvious advantage of the Assembly Line is that it breaks large projects into smaller, easier to manage, units of work. As products are designed and assembled, they can be inspected after each stage of work to assure the work product conforms to the level of work effort. If not, it can either be corrected or rejected, thereby inconsistencies and oversights are detected and corrected prior to progressing to the next phase of work. The Assembly Line, therefore, promotes quality and consistency of workmanship.

Because of the magnitude of their Customer System, Kansai Electric felt uncomfortable with using traditional methods for implementing the project; e.g., superficial requirements definition and "trial and error" programming. Instead of relying on the intuition of programmers, as they had done in the past, they devised a scientific method. Historically, programmers had been allowed free-reign in terms of their approach to design and solve problems. This led to inconsistent and incompatible results, and ultimately the disarray of the existing Customer System.

As Kansai redefined its development environment, it specified the concepts and principles it would adhere to during development. This included a definition of terms. With this conceptual foundation in place, they were then able to define an assembly line process to develop the new Customer System.

Kansai's assembly line specified the sequence by which resources were defined and related to form products. For example, in Phase 1 information requirements were defined and related to the data elements needed to support them (both new and existing). In Phase 2, the requirements were related to the business processes (sub-systems) and outputs (screens and reports) implementing the information; further, the sub-systems were defined and related to the inputs, outputs and files associated with each business process. In Phase 3, the procedural work flow of a business process was defined and related to inputs, outputs and files. Ensuing phases were used to decompose the system into finite detail with pertinent resource relationships. When it came time to write software for the programs, everything was explicitly defined and documented, thereby taking the guesswork out of programming.

Such a disciplined development environment is useful as a means for improving communications. Because the methodology is precisely defined, an effective dialog can be established between developers to distinguish Who is to perform What work, When, Where, and Why. This was vital for a project as massive as the BEST Project.

The BEST Project brought together a team of more than 200 analysts and programmers from the four banks. Recognizing project assignments would require a mix of personnel on the various stages of work, Project Management defined an assembly line process which defined the activities and work products for each stage of work. As a result, each member of the project understood their duties and responsibilities and the type of results expected from each stage of work.

Each phase was also defined in terms of the skills and proficiencies required to perform the activities. Consequently, the BEST Project Managers found they could plug people in and out of the development process without having an adverse effect on the project schedule. On most occasions, an analyst or programmer was assigned a specific phase of work to carry through to completion. However, in order to balance the workload of the staff, it was common for Project Management to reassign a worker from one phase to another. For example, an analyst or programmer would suspend activity at one point of the project, transfer to another phase of work, and allow another person

to complete the phase without losing a significant amount of time. Further, because the specifications for the work products were well defined, it was easy to suspend work in one phase and resume it at a later date.

An assembly line process, such as the methodologies used by Kansai Electric and the BEST Project, is an effective means for managing human resources to maximum effect. Such personnel mobility is unheard of under traditional methods for systems development. Traditionally, if a phase is suspended, all design specifications must be recreated before proceeding with the phase.

The methodologies used by Kansai and BEST exhibit the following assembly line characteristics:

- \* Standard and re-usable Work Breakdown Structures (WBS) and precedent relationships. The WBS represents various levels of work effort, from general to specific. Such a breakdown is needed to decompose project work into manageable pieces. Re-usable structures provides for uniform work effort and work products. Consequently, workers do not have to be re-trained when asked to execute the same type of work step in another project. For example, someone well versed in performing a "System Design" phase need not be re-trained when asked to perform a "System Design" phase in another project.
- \* Precedent relationships specify the dependencies between steps in the WBS (preceding/succeeding work steps in a project). This is required to specify the direction of the work flow in a project. Without it, projects cannot proceed from one step to another. Such dependencies must allow for both parallel and sequential execution of projects.
- \* Re-usable Work Breakdown Structures provide a logical consistency between projects. As such, it provides a means for establishing metrics between projects and allows management to make a comparative analysis between projects and workers.
- \* Defined work products (deliverables) for each work step in the WBS. Deliverables substantiate completion of work and adherence to the methodology. The work product is either produced or it is not. The phase of work remains open until the work product has been produced.
- \* Defined review points and acceptance criteria for design reviews. This provides the means to evaluate the quality of work and either approve the work product, request revisions, or reject the work product outright.

Bottom-line, the assembly line process can turn a heterogeneous development environment into one that is homogeneous. By demystifying the development process, the project teams within Kansai Electric and the BEST Project were able to communicate at a common level, both internally within the staff and externally with end users. This, in turn, promoted cooperation and teamwork from all parties involved.

## **DIVISION OF LABOR**

Coupled closely with the Assembly Line concept is the Division of Labor, the purpose of which is to break the production process into separate tasks performed by specialists. Defining the relationship between work steps and the development function delineates the duties and responsibilities within a project. For example, a Systems Analyst is responsible for performing the System Engineering phases, and a Programmer is responsible for performing the Software Engineering phases.

Each phase of work can also be defined in terms of the required skills needed to perform the work along with the required proficiency. This is useful for determining staffing and training requirements for a project.

Historically, programming has represented the lion's share of work effort in a systems development project. Unfortunately, this leads to the "trial and error" method of development as mentioned earlier. In this type of environment it is not uncommon to find three programmers for every systems analyst (4:1 or 5:1 is also quite common). Because the orientation of IRM is on engineering total systems and not just on programming, this trend should be reversed.

### **Ajinomoto Company, Ltd.**

Ajinomoto is the world's leading manufacturer of amino acids and a diversified food producer with total sales of \$5 billion. In the early 1980's the company found its systems development organization in disarray. There were no formal methodologies in place, standards were rare, and organizationally the department was in chaos. Like so many other development organizations, Ajinomoto had a 3:1 ratio of programmers to systems analysts and found they were spending considerable time in programming (approximately 80% of a project). Instead of specifying information requirements and engineering an overall system architecture, Ajinomoto's development staff was spending most of their time writing computer software. Because the software rarely satisfied user requirements, the staff spent considerable time re-writing programs. As a result, the company found itself in a constant "firefighting" mode and little progress was made towards achieving the company's major goals.

During the mid-1980's, the company re-configured their development organization by introducing a mass production environment similar to the ones created by Kansai Electric and the BEST Project. Emphasis in the new environment was on design correctness (producing systems that accurately satisfy user information requirements) and building enterprise-wide systems. In addition to implementing a regimented assembly line process, the company realized it would be necessary to retrain the development staff to learn the methodology and develop the types of skills needed to implement major systems. As the shift in orientation from software to total systems gradually took effect, Ajinomoto began to experience project teams consisting of only 31% programmers. Because of the emphasis on up-front planning and design, the

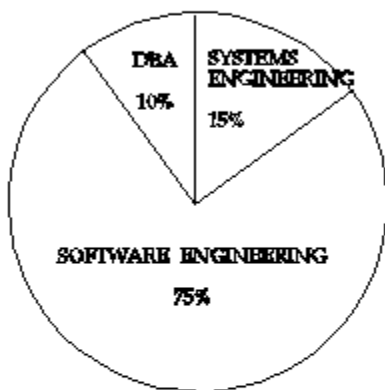
programming phases of Ajinomoto's methodology have dwindled to only 21% of the overall development process (with less than 5% of the time spent on coding). Kansai Electric and the BEST Project have reported similar experiences.

The "trial and error" approach to systems development is based on the belief that programming represents the most significant work effort in the development process. Users and developers alike do not believe they are being productive until they begin software engineering. Consequently, an insignificant amount of time is spent analyzing the business, defining information requirements, determining business processes, and specifying software requirements. This approach results in an inordinate amount of time in programming with questionable results (e.g., systems and software that do not meet user needs; elegant technology addressing the wrong business problems).

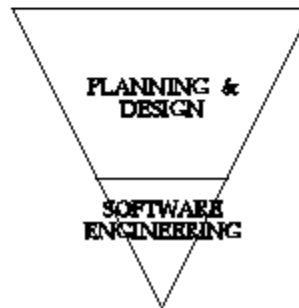
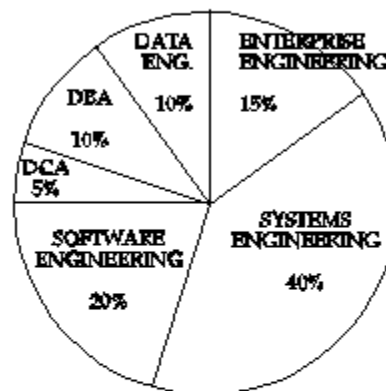
A true IRM environment represents the antithesis of the "trial and error" approach. Considerable time is spent up-front studying problems, specifying requirements, and developing the overall system architecture. As a result, software engineering is de-emphasized and represents only a fraction of the overall development process. In the IRM environment, software specifications are defined with a high degree of precision, thereby eliminating misinterpretations and a lot of re-programming.

## ***DISTRIBUTION OF WORK***

***TRADITIONAL APPROACH***



***IRM ENVIRONMENT***



## STANDARDIZATION OF PARTS

The Standardization of Parts is required in Mass Production for interchangeability and assembly by unskilled or semiskilled workers.

As simple as it may sound, concepts such as sharing and re-using resources can offer a company considerable leverage, as demonstrated by the MRP analogy. It is difficult to imagine a company designing and manufacturing a line of automobiles without standard parts. Under this scenario, each model would be designed with a distinctively separate set of parts. Because of the lack of conformity and standardization, it would not be possible to share and re-use parts between models. Further, inventories would increase exponentially to accommodate the increase in total parts. Obviously this is not a practical way of operating. Standardization of parts is an inherent part of engineering and manufacturing and is a requirement for Information Resource Management.

There is more to an information system than program source code. Much more. An information system consists of many processes, inputs, outputs, files, records, data elements, etc. We recently conducted a research study of the number of information resources and design decisions in several of our customers' information systems. The purpose of the study was to determine the scope of systems being developed today. Many diverse systems were analyzed from a cross-section of industries and applications, everything from small programming assignments to major systems such as those developed by Kansai Electric and the BEST Project.

One of the key observations made in the study was that there is a finite number of design decisions associated with each type of information resource. As an example, for an output, decisions have to be made as to its physical media (screen or report), size (number of characters), messages associated with it, etc. For a data element, its logical and physical characteristics must be specified (definition, type, size, class, length, etc.). For a program, the language to be used, software logic, required file structures, etc. These design decisions can be simple or complex; regardless, they are all required in order to design a system. When we multiply the number of design decisions by the number of information resources in a system, we get an idea of the magnitude of the average information systems development project:

	AVERAGE NO. RESOURCES IN AVERAGE SIZE SYSTEM	NUMBER OF DESIGN DECISIONS	DECISIONS IN AVERAGE SYSTEM DESIGN
	_____	_____	_____
SYSTEMS	1	25	25
SUB-SYSTEMS (business processes)	15	25	375

PROCEDURES (both computer & administrative)	40	30	1,200
PROGRAMS	75	30	2,250
OPERATIONS (manual steps)	125	10	1,250
INPUTS (interactive or batch)	50	15	750
OUTPUTS (screens & reports)	200	15	3,000
FILES (logical and physical files, computer and manual)	100	30	3,000
RECORDS (includes file structures, print maps, panels, input transactions, etc.)	1,000	30	30,000
DATA ELEMENTS	400	20	8,000
TOTAL NUMBER:	2,006		49,850

**NOTE:** Decisions are design oriented only; they do not include Project Management related decisions (such as those associated with planning, estimating and scheduling).

## RESOURCE CHARACTERISTICS

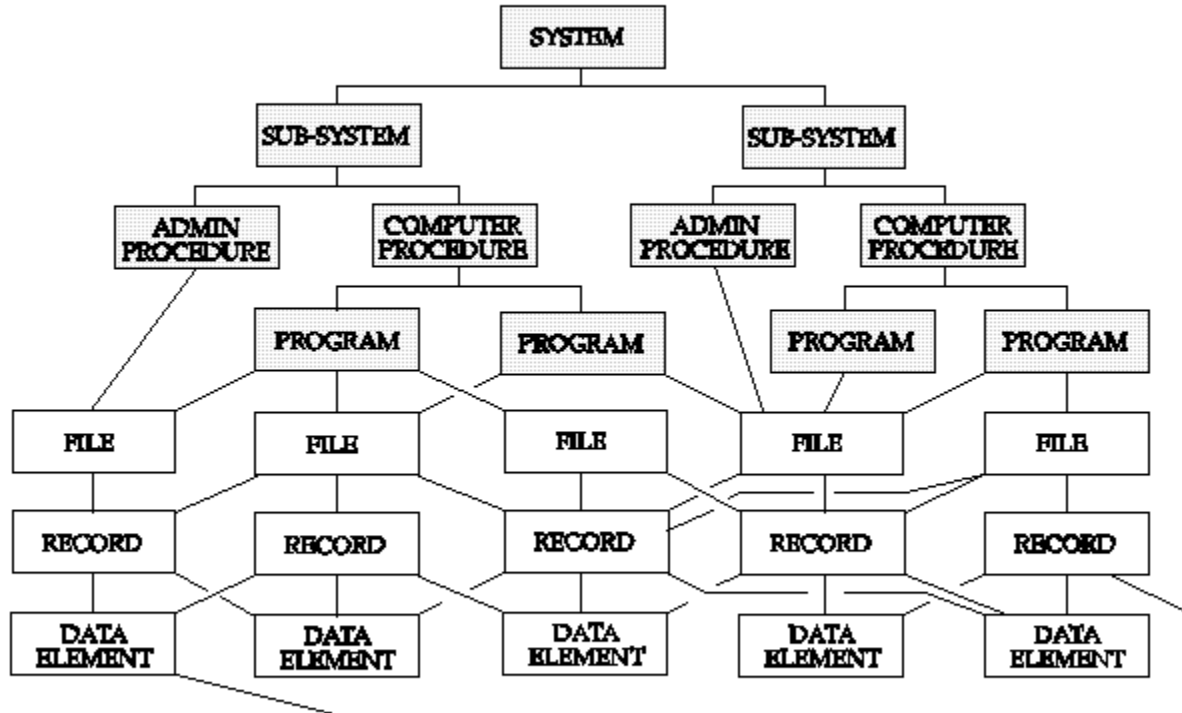
You cannot share and re-use a resource if it is not documented. Having a tool to track the location of information resources may be useful for inventory control purposes, but it does little to detect redundancies. In order to share and re-use resources, the basic building blocks have to conform to a prescribed criteria, thereby assuring their completeness and conformity to standards. Knowing this, Kansai Electric used a sophisticated tool to catalog and classify resources. Again, borrowing a lesson from engineering/manufacturing, Kansai made use of a "Bill of Material Processor" (BOMP) specially adapted for managing information resources.

In manufacturing, a BOMP is a tool to classify and track the parts of a product and how they interrelate. A similar tool can be used to inventory and standardize information

resources.

# ***IRM RESOURCE RELATIONSHIPS***

***A 'BILL OF MATERIALS' FOR AN INFORMATION SYSTEM***



***RESOURCES ARE IDENTIFIED BY NUMBER AND NAME.  
THEY ARE ALSO SHARED AND RE-USED THROUGHOUT SYSTEMS.***

As part of their re-evaluation of the development process, Kansai Electric made BOMP the cornerstone of their development efforts. As analysts and programmers progress through a development project, information resources are cataloged and cross-referenced to other resources in Kansai's BOMP. In addition to identifying each resource by number and name, the characteristics of each resource is described. For example, an information requirement is defined in terms of its required timing (Frequency, Offset, and Response Time), the type of information it represents (Policy, Control, or Operational), and a text description. Data elements are defined in terms of their logical and physical attributes (type, class, size, length, source, precision, scale, program labels, etc.). These attributes become the basis for detecting and eliminating resources with redundant characteristics.

A project as large and encompassing as the BEST Project would not be possible without a BOMP-like tool to catalog and control the thousands of resources and design decisions associated with it.

Another top Japanese bank, Norin Chuo Kinko, was able to revitalize an aging banking system by implementing a mass production environment using a BOMP tool to catalog



and control its information resources. This one application supported branch offices in 36 cities throughout Japan and involved 3,740 programs (over 2.8 million lines of code) supported by a Data Base Management System. Integration of the business, systems and data resources would not have been possible without a BOMP tool to record the components and their relationships.

Although the principal purpose of BOMP is to control and classify resources, it has the added capability of performing an "impact analysis" on resources, which is an analysis of resource relationships; for example, if one resource is changed, "impact analysis" will list the chain of resources that are either directly or indirectly affected. Such capability provides the ability to study and control changes.

As the BOMP is populated with information resources, other development aids (such as software engineering tools) can tap into the BOMP and download vital design intelligence for producing software and data base structures.

This leads to the final element of Mass Production...

## **PRECISION TOOLING**

The purpose of Precision Tooling is to provide mechanical leverage for performing tasks routinely with a high degree of accuracy and uniformity. Whereas an Assembly Line process specifies the 5-W's (Who, What, When, Where, Why), tools implement specific techniques which dictate "How" various tasks are to be performed.

In a Mass Production environment it is important to understand when to use the right tool to perform the right task. Failure to understand this will result in the wrong tool being used under the wrong circumstance which, obviously, will be counter-productive. To illustrate, industrial robots offer an efficient means to implement mundane tasks such as welding. However, if a weld is performed at the wrong time and in the wrong place, efficiency becomes an irrelevant issue.

Defining the Assembly Line is a precursor to the selection and deployment of tools. If the development environment is not defined properly, supporting tools will inevitably be misapplied and misused. To illustrate, consider the misapplication of Computer Aided Software Engineering (CASE) tools in the United States. Studies have shown that as little as 25% of all CASE tools purchased to date have been effectively implemented. As a result, the advertised benefits of these tools are seldom realized due to their misapplication.

In contrast, several Japanese companies have been able to effectively capitalize on CASE technology, simply because they knew where the tools are to be used in their defined development environments. For example, Kansai Electric maximized the use of Software AG's NATURAL fourth generation language (program generator) and ADABAS data base management system (DBMS) in the new Customer System.

There have been other similar examples:

- \* Kibun Company, Ltd. is a large manufacturer/processor of seafood, including today's popular artificial crab meat. The company has also developed a reputation for building perhaps the best "JIT" (Just In Time) system in Japan. Like Kansai Electric and the BEST Project, Kibun has implemented a disciplined development environment featuring defined phases of work. During the software engineering phases, Kibun makes active use of the TELON program generator from Computer Associates.
- \* JGC Corporation is a major engineering/construction firm headquartered in Tokyo. JGC has been using program generators for nearly a decade. This has significantly changed the focus of the development staff away from writing software, and towards engineering total systems.
- \* Yokogawa Electric Corporation is a prominent manufacturer of electronic components and instruments. Their development environment includes the use of a UNIX toolkit during the software engineering phases.
- \* Kyowa Hakko Kogyo Co., Ltd. is a manufacturer of solvents spirits, and yeast. Their environment makes use of the POWERHOUSE fourth generation language from Cognos Corporation.

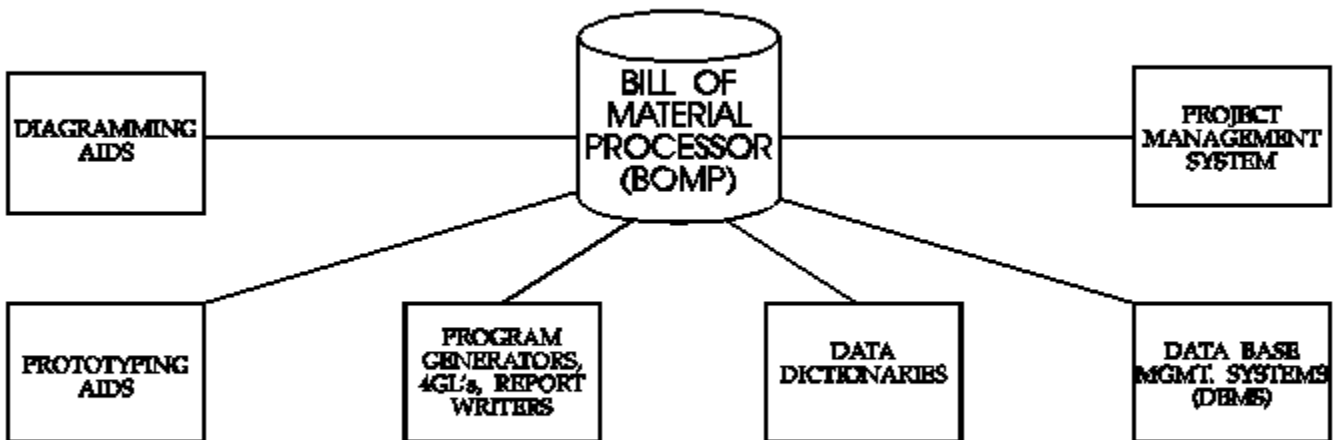
All of these companies were able to realize the full potential of the tools simply because their development environments were well defined and they knew precisely when and where to use the various tools. As a result, they can plug tools in and out of the development environment as required.

Today, there are so many application development aids available that it is not uncommon to find a single development organization using a multitude of CASE tools. Further, there is no single vendor with a complete "womb to the tomb" application development aid for some rather obvious reasons: companies use different programming languages, design techniques (structured programming versus object oriented programming), input/output techniques (graphical user interfaces versus full screen menus versus command languages, etc.), and file management techniques (including various DBMS architectures), and operating systems. Computer technology is simply evolving too rapidly for a single vendor to offer a single comprehensive product that does everything. Instead, companies must orchestrate a variety of tools to work in a concerted manner. Unfortunately, the various tools were not designed to be compatible. Different CASE vendors implement different techniques using different concepts and terminology. For example, a "file" in one product may be called a "data store" in another product, and a "table" in another. Such anomalies inevitably create confusion. Even when the same terminology is used, they often express different meanings.

Fortunately, tool integration is made possible through Standardization of Parts as implemented using a Bill Of Materials Processor. When implementing a new tool, a

company must translate the vocabulary of the tool to the standard parts as maintained by the BOMP. Consequently, the tool can access the intelligence regarding the various information resources maintained in the BOMP. CASE tools can either export design intelligence from the BOMP or be used as a vehicle to import information resources to the BOMP.

## ***BOMP/TOOL RELATIONSHIPS***



***BECAUSE THE BOMP IS THE FOCAL POINT FOR RESOURCE DEFINITIONS, IT CAN INTERFACE WITH A VARIETY OF DEVELOPMENT TOOLS***

Interfacing tools is not as difficult as it may seem. All of the companies mentioned above have interfaced their various CASE tools to their in-house BOMP. As a result, the tools are synchronized and work in harmony.

### **THE NEED FOR INDUSTRIAL ENGINEERING**

Implementation of the five elements of mass production for designing and managing information resources is no small task. Ultimately, it represents implementing an "Information Factory" environment complete with assembly lines, production control (to monitor project status) and materials management. Fortunately, there is a discipline oriented to this task, again, derived from engineering and manufacturing: Industrial Engineering (IE).

The purpose of the IE function is to define the mass production environment in terms of:

- \* Defining the stages of work effort as per the Assembly Line.
- \* Defining the necessary skills and proficiencies required to perform the work.

\* Preparation of job descriptions detailing the duties and responsibilities for the various work functions.\* Evaluation and selection of pertinent tools and techniques to be used on the Assembly Line.

This is not a one-time endeavor. In fact, Industrial Engineering is an on-going process which monitors the environment and constantly seeks new ways to improve production. The Industrial Engineer uses work sampling and work measurement techniques in gauging performance. In many instances, IE fulfills the role of inspector to evaluate the quality of work products.

All of the Japanese companies mentioned in this article have gone through this type of IE process in order to implement their new development process. This represented a considerable re-orientation, re-training, and re-tooling effort by the various companies. The work place had to be re-configured, attitudes towards systems development had to be modified, and new skills had to be learned.

On the surface, it would appear such an environment would be more suited towards engineering/manufacturing related companies, who could easily assimilate such concepts, as opposed to service related industries (e.g., banking, insurance, etc.). However, the fact that companies such as those involved in the BEST Project, and the Norin Chuo Kinko bank indicates that mass production is adaptable to the service sector as well.

## **THE RESULTS**

### **Kansai Electric**

After implementing their new environment, a project was initiated to replace the aging Customer System. Three years later, the first stage of the project was delivered on time and within costs. Stage one primarily represented the base operational needs of the Customer System, and included 858 programs (with over 829K lines of source code) in an integrated data base environment. The system now processes over 300K transactions per day and supports over 2,410 end users, such as the Customer Service department, Billing, and Accounting.

Since its initial implementation, the Customer System has progressed into additional stages of development to integrate with other corporate systems and incorporate additional features without replicating or supplanting existing information resources. The ensuing stages will interface the Customer System with Banking Systems, Energy Load Projections/Balancing, and Strategic Planning.

### **The BEST Project**

From inception to completion, the BEST Project lasted three years. In that time, the project produced over 70 major integrated systems. Upon completion, the project

members dispersed and the systems were implemented by the four banks. Although it was originally planned that a group of 50 analysts and programmers would remain to correct bugs, after six months it became apparent such a large group would not be necessary due to how well the systems were performing. Subsequently, the group was reduced to 25 and eventually phased out completely.

### **Ajinomoto Company, Ltd.**

Since 1989, when the company re-configured its development organization, it has produced four major systems for sales, accounting, research and production. These systems run on multiple computer platforms and consist of more than 2,650 programs. The systems have greatly reduced inventory overhead, led to "Zero Defects" in operations, and cost reductions of 400 million yen per year (\$3.4 million).

As part of their on-going analysis of the environment, the company discovered a direct correlation between user participation in the early development phases to the quality of the system and user satisfaction with the finished product. This would not have been possible if the environment was not defined and the duties and responsibilities of both developers and end-users alike were not specified and understood by the participants.

### **CONCLUSIONS**

As the Japanese have demonstrated, moving IRM from an art to a science can make a significant contribution to the overall productivity of a company. What is interesting in the Japanese case studies is that defining an IRM environment did not require any new theories of management, but rather, the use of basic common-sense techniques that have existed for years. Instead of arguing over the semantics of IRM, the Japanese have been able to adopt a pragmatic solution.

Moving IRM from an art to a science has its advantages. It demystifies and simplifies the development process. As a consequence, it can be easily taught and managed in a consistent manner which provides a means to produce more systems in a uniform manner. Critics might argue a "science" inhibits individual creativity. Nothing could be further from the truth. Disciplines such as engineering, architecture, even music, are all regarded as sciences, yet have some of the most creative people in the world. A "science" simply establishes the governing rules. Any company still hand-crafting each system is doomed to extinction. The demand for information in today's business world mandates the use of a scientific method using mass production concepts.

The Japanese case studies also highlight how they perceive the concept of productivity. Whereas most western companies are obsessed with "efficiency," the Japanese are equally concerned with "effectiveness." Whereas efficiency addresses how fast a task is performed, effectiveness addresses the necessity of the task itself. Again, using the industrial robot example mentioned under "Precision Tooling," the Japanese feel there is nothing more unproductive than to perform something efficiently that should never have been performed in the first place. Productivity, therefore, is viewed as:

## **PRODUCTIVITY = EFFECTIVENESS X EFFICIENCY**

Although a program generator offers efficiency in writing software, if the program is not needed or doesn't serve a business purpose, it is useless. 100% efficiency multiplied by 0% effectiveness equals 0% productivity.

Because the Japanese were able to communicate the IRM concept in practical business terms, such as the MRP example, they were able to successfully recruit support from upper management. If presented as a cryptic concept, management would have simply regarded it as another form of computer chicanery. Instead, the Japanese believe information resources can be developed and managed like any other corporate resource (e.g., materials, human, equipment, financial resources). The only difference is that IRM deals with a much less tangible resource.

Although there are some short-term benefits associated with IRM, the real benefits are long-term in nature. Sharing and re-using resources is an evolutionary process. Resources must be carefully designed and standardized before they can be re-used. Consequently, IRM requires executive visionaries with an eye on the future. There is an old saying in Japan that perhaps sums up the IRM concept best, *"You must plant the seed before you can harvest the crop."*

**END**

**COPYRIGHT (C) MBA 1995**

**NOTE:** *This article first appeared in the May 1987 issue of MBA's Management Visions newsletter. A modified version of the article appeared in the October 1987 issue of INFOSYSTEMS.*

## ***THE FOUR STAGES OF IRM GROWTH***

It seems that everyone is aspiring to perform Information Resource Management (IRM), managing the resources by which information is produced; but is anyone actually doing it? Most people don't even understand the problem, let alone how to build an effective IRM environment.

How does a company grow into an IRM environment? Does it happen overnight? Hardly. It evolves slowly over time as companies become aware of the role of information in their organization. For most companies, it is a costly and painful learning process. The following is a description of the four stages of maturity that companies typically follow: Birth, Childhood, Adolescence and Adulthood.

### **BIRTH**

The day a company goes into business is the day when its information systems are born. When a new company or organization is established, there are some very primal information requirements to accommodate the operation of the enterprise. For example, basic bookkeeping (billing, payroll, government reporting, etc), minutes of meetings, recording of policy decisions, correspondence, etc.

To implement these basic administrative requirements, simple office equipment is all that is required, such as typewriters, dictation equipment, calculators, photocopiers, telephones, etc. An Office Manager with a clerical staff (e.g., secretaries, book-keepers) normally implements these processes and operates the equipment. During this stage, their concern is for implementing basic manual procedures with an eye for work simplification to minimize overhead.

As the business expands and becomes more complicated, whether from an increase in employees and/or business, there is a growing demand for more information which leads to the next stage of growth...

### **CHILDHOOD**

This stage is entered either by an emerging company or an established firm that is pressured to investigate the potential of new technology, namely the computer, to give leverage to their business needs. This is a stage which most of the "FORTUNE 500" companies and major government institutions went through in the 1950's, 60's and 70's.

In the childhood stage, the intent is to investigate the potential of the computer. This is

an age of experimentation where a highly complicated and technical device is introduced to a company. This new technology, of course, requires a technically oriented individual to operate it. Someone who is more in tune with the equipment, as opposed to the problems and objectives of the enterprise.

The computer is typically centralized in one location until someone can determine an appropriate way to apply it to the business.

This stage results in the executive's "black box" image of the computer. As a consequence, they divorce themselves from the machine, and appoint a "Data Processing Manager" who is given free reign over the new technology. Like the staff that supports him, the D.P. Manager is technically inclined (probably just one step ahead of a programmer).

The "Data Processing Department" tackles simple problems aimed at automating some of the basic administrative routines of the company. There is not considerable pressure to satisfy business problems, only a "see what you can do" type of attitude. As a result, the D.P. staff takes an ad hoc, "quick and dirty" programming approach to problem solving. This type of philosophy sows the seeds for problems to come in the years ahead. For example, applications are not integrated, data is not shared (data redundancy is commonplace) and documentation is non-existent, applications are not easy to maintain or modify. As a result, they are constantly being discarded and rewritten, further compounding the problem.

One of the most significant aspects of this stage is that it fosters the "tool oriented approach" for solving problems. The attitude of the staff is that the only legitimate problems worth solving are those that can be addressed by the computer. All others are immaterial. This is a frame of mind that will take considerable time to overcome. The indifferent attitude of the D.P. Department irritates and alienates end users who have increasing demands for information.

Impatient for results, management begins to apply pressure on the D.P. Manager for more applications to satisfy user demands. This forces the next stage ...

## **ADOLESCENCE**

This is the age of awakening for most companies, an era when the D.P. Department begins to manage itself in order to accommodate growing business demands. The D.P. Manager is supplanted by an "MIS Director," someone who is a little more adept at management politics. In fact, the "MIS" (Management Information Systems) designation is indicative of the company's growing awareness of the need for information.

In this stage, the MIS Director implements rudimentary management controls, particularly in the areas of project management and documentation. Using the "tool oriented approach" to improve staff productivity, the MIS Director implements several



software tools and techniques, such as: Data Base Management Systems (DBMS), Data Dictionaries, Program Generators, Report Writers, Fourth Generation Languages (4GL), Test Data Generators, Computer Aided Software Engineering (CASE) workstations, Structured Programming, etc.

Dazzled by sophisticated software and in fear of "falling behind" in the technology race, the MIS Director authorizes the purchase of tools that implement esoteric (some prefer to call it "Voodoo") management principles.

Unfortunately, the MIS Director is seduced and abandoned by the technology; the results are still the same: Applications do not satisfy user needs, applications are not integrated, data redundancy is still pervasive, applications are still difficult to modify and maintain, and the staff remains a free-spirited group of technicians.

The "tool oriented approach" is very costly to the company, but the results are still the same. The MIS Director is still supported by a technical staff that believes that the "real work" is in the production of software, where their programming skills excel. The "Analyst/Programmer" is really nothing more than a senior programmer.

Superficial standards and pseudo-scientific management techniques are applied to the development process. An application project typically consists of the classical approach for developing systems: A primitive Feasibility Study, General Design (sometimes referred to as "External Design"), Detail Design ("Internal Design"), Programming (usually following a Structured Programming Guru's technique), Testing, Installation, and Review. In this situation, programming remains 85% of the entire project. This approach is usually well packaged in voluminous standards manuals (which no one but the DP Auditors read).

The computer is decentralized with terminals, minis and micros being distributed throughout the company.

The end User, who is frustrated by the lack of support from the MIS Department, turns to the Personal Computer (PC) for help. Unfortunately, the User is no more adept at using the computer as the MIS people are in using the mainframe and the problems are compounded even further (particularly in the area of redundant data).

Despite the substantial investment in computer hardware and software thus far, management finally recognizes that conditions are intolerable and that the company is not getting a satisfactory return on investment. This becomes the catalyst for change. Without it, the company stagnates and the situation worsens. Adolescence must eventually give way to ...

## **ADULTHOOD**

This stage represents a radical departure from the past mode of operation. Very few companies, if any, have reached this stage of growth yet. It represents a mature

environment where the information systems staff is in tune with the mission of the company, and information is viewed as a corporate asset used for strategic purposes. This is the age of Information Resource Management (IRM). This philosophy gives rise to the Chief Information Officer (CIO), a legal officer of the company, not just a job title.

No longer is the "tool oriented approach" pervasive in the company. It was tried, and it failed. The latest "state of the art" technology is a worthless status symbol if it doesn't contribute to the profitability of the company.

Now, the CIO turns to tried and proven approaches to management. Information Systems design is no longer viewed as an art, but a science. The CIO organizes the systems development environment into an Engineering/Manufacturing company, complete with Assembly Lines, Production Control and Materials Management. As a result, the systems staff is transformed from free spirited programming "hackers" to a group of disciplined and quality conscious business professionals. In some respects, the staff will resemble the "Methods & Procedures" staff of yesteryear who had a business orientation.

The computer is viewed as just another piece of office equipment; they are not discernible. Users and management no longer fear technology because the CIO implements it effectively into the business. In the adult stage, the emphasis is on complete and integrated information systems, not just software. Programming is less than 15% of the entire development process, with the bulk of the work being expended on business analysis. Data is managed as a resource and redundancy is eliminated. All of the problems experienced earlier disappear.

As enticing as adulthood may sound, very few companies have the management skill or fortitude to make it happen, particularly in the United States. Most companies don't even understand the problem. Adulthood represents a substantial and long-term corporate commitment, not just departmental commitment, which most American companies strongly resist. Instead, they are content with short-term "quick and dirty" solutions. On the other hand, Japanese companies, who are much more far-sighted, have a greater chance for success and are rapidly moving into the adulthood stage. This will make them increasingly more competitive in the years ahead.

## **SUMMARY**

Over the last ten years alone, computer technology has changed radically, job titles and terminology have changed, and salaries have risen sharply, but little else has changed. The information problems of today are no different than 10, 20 or 30 years ago. Despite today's technology, companies still experience:

- \* Project cost overruns and slipped schedules.
- \* Poor communications and relations with the User community.

- \* Redundant data and lack of application integration.
- \* Applications are difficult to modify and maintain.
- \* Lack of adequate documentation.
- \* Design inconsistencies.
- \* Applications still do not satisfy User needs.
- \* Hardware/Software dependencies.
- \* Employee dependencies to maintain systems.

The tools and characters have changed, but the tune remains the same. Regardless of the titles and technology used, most companies in North America are stuck in either the "Childhood" or "Adolescent" stages of growth. Indicative of this are the journals, trade groups, universities, and trade shows that still promote the "tool oriented approach" as opposed to promoting management. Systems Development is still viewed by many people as an art, not a science. Whereas, it is a science in reality. It has established and proven concepts and can be taught as a science.

***No amount of elegant technology will solve our problems, only strong management will.***

THE FOUR STAGES OF IRM GROWTH

CHARACTERISTICS	BIRTH	CHILDHOOD	ADOLESCENCE	ADULTHOOD
INFORMATION Age of IRM.	Concern for	Experimentatn.	Awakening.	
ENVIRONMENT Strong Mgmt.	manual	'Quick &	Applying	
Science vs	processing;	dirty pro-	rudimentary	
Art.	Work simplifi-	gramming.	management	
Discipline,	cation	Beginning of	techniques &	
Organization		the "tool	tools. Pro-	
& quality		oriented"	gramming is	

conscious.		approach.	85% of effort
------------	--	-----------	---------------

-----+-----+-----+-----  
+-----

APPLICATIONS Information as asset and strategic weapon	Basic book- keeping and clerical tasks.	Program basic administrative routines	Major systems, but lacks integration.
--	--	---	---

-----+-----+-----+-----  
+-----

EQUIPMENT Computers blend in with office equipment.	Basic office equipment	Centralized computer	Decentralized computing
--	---------------------------	-------------------------	----------------------------

-----+-----+-----+-----  
+-----

PERSONNEL C.I.O.reporting to C.E.O. with business oriented staff	Office Manager and clerical staff	D.P. Manager reporting to administra- tive area.	MIS Director & Programmer/ Analysts
--	---	---	---

-----  
-----

**END**

**COPYRIGHT (C) MBA 1995**

**NOTE:** *This article first appeared in the May 1987 issue of MBA's Management Visions newsletter.*

## ***CORPORATE CULTURE***

The subject of "corporate culture" seems to be on everyone's mind these days; from the college graduate entering the job market, to the IRM executive who is trying to improve management and productivity in his organization. It is the topic of interest at social as well as professional gatherings.

The perceptive manager understands the importance of establishing and controlling the work environment, including both logical and physical considerations. Unfortunately, many managers don't appreciate the concept of corporate culture and how to use it to their advantage.

Corporate culture pertains to the identity and personality of the company we work with, either in the private or public sectors. All companies have a culture; a way they behave and operate. They may be organized and disciplined or chaotic and unstructured. Either way, this is the culture which the company has elected to adopt. What is important is that in order for an employee to function and succeed, they must be able to recognize, accept and adapt to the culture.

### **MEMBER VS. ALIEN**

Have you ever noticed how people react to foreign visitors; whether an exchange student or a visiting professional? The stranger may be welcomed, but may never be accepted unless that person can adapt to the norms of their new environment. If they don't, the members will shun the stranger and reject the alien from their culture. The same is true in business. If the new employee, consultant or visitor cannot adapt to the corporate culture, their chances for success are slight. The members of the culture will reject the person outright and will work against them.

The reason for this phenomenon is that people tend to prefer conformity in their culture. Conformity represents a harmonious environment where the behavior and actions are predictable. Most people have a deeply rooted desire for a sense of order and stability in their lives, which is what conformity provides. A stable environment promotes self-confidence in the members of the culture and allows them to concentrate on their work.

### **HUMAN PERSPECTIVE**

Corporate culture deals with how we see ourselves and others. We act on our perceptions, not necessarily what occurs in reality. The culture greatly influences our perspectives and behavior. For example, our values and beliefs may distort what happens in fact. Gossip, propaganda, and a sensational press, deals with what people

want to hear, not necessarily what happens in fact.

## **DEFINING CULTURE**

Before we can alter the culture, we must first understand it. Culture is defined as the characteristics of the members of a civilization. Ultimately, culture defines the quality of life for a group of people.

Culture doesn't appear suddenly, it evolves over time as people grow and learn. The older the heritage, the more ingrained the culture is in its members.

There are essentially three parts to any culture: Customs, Religion and Society. Each influences the others.

**1. CUSTOMS** - Webster defines custom as a *"long-established practice considered as unwritten law."* Custom dictates the expected manner of conduct for the culture. It prescribes the etiquette to be observed in dress, speech, courtesy and politics (gamesmanship). Several companies, most notably IBM, have long understood the power of customs. These norms are established to project a particular image the company wishes to convey.

**2. RELIGION** - Religion is the philosophy of life and the basis for our values. It influences our judgement in terms of what is ethical and what is not. Although uniform morality sounds attractive to executives, it can be quite dangerous if unethical practices are allowed to creep into the moral fiber of the company.

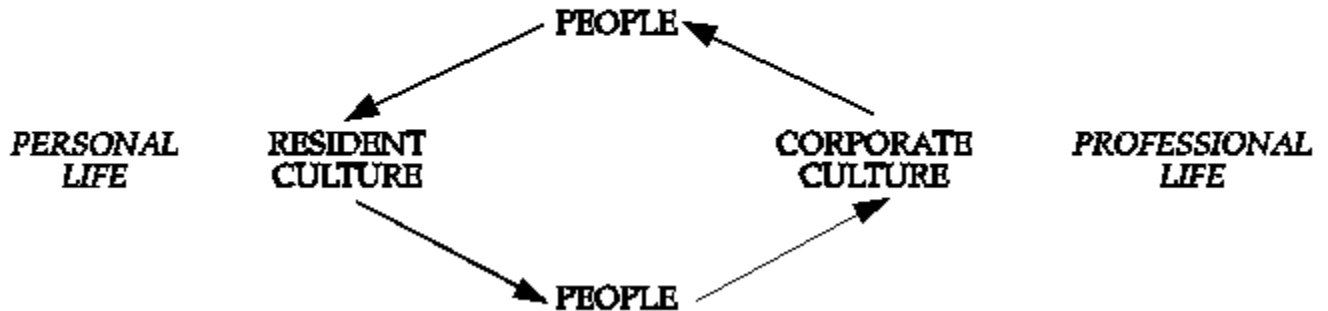
**3. SOCIETY** - Society defines our interpersonal relationships. This includes how we elect to govern and live our lives. Society defines the class structure in an organization, from Chairman of the Board to the hourly worker. It defines government, laws and institutions which must be observed by its members. More often than not, the society is "dictated" by management as opposed to "democratically" selected by the workers.

## **INFLUENTIAL FACTORS**

Obviously, it is people, first and foremost, that influence any culture. In terms of corporate culture, the only external factor that influences the enterprise is the "resident culture," which is the culture at any particular geographical location. The resident culture refers to the local customs, religion and society observed in our personal lives, outside of the workplace. The resident culture and corporate culture may differ considerably in some areas but are normally compatible.

Anthropologists have long known that the physical surroundings, such as geography and climate, greatly influence the resident culture. The resident culture, in turn, influences the corporate culture. The corporate culture, which affects the behavior of its members, will greatly influence the resident culture.

# **CULTURE IS LEARNED**



## **SUB-CULTURES**

Within any culture there are those people exhibiting special characteristics that distinguish them from others within an embracing culture; this is what is called "sub-cultures." In a corporate culture, sub-cultures take the form of cliques, special interest groups, even whole departments within a company. This is acceptable as long as the sub-culture doesn't violate the norms of the parent culture. When the characteristics of the sub-culture differ significantly from the main culture, it becomes a culture in its own right. This situation can be counterproductive in a corporate culture, a company within a company. For example, we have seen several IS organizations who view themselves as independent of the companies they serve. They "march to their own drummer" doing what is best for the IS Department, not necessarily what is best for their company. Conversely, we have seen management regulate the IS department as a separate, independent group as opposed to a vital part of the business.

## **CHANGING THE CORPORATE CULTURE**

Changing the corporate culture involves influencing the three elements of the culture: Customs, Religion and Society. This is not a simple task. It must be remembered that culture is learned. As such, it can be taught and enforced. However, the greater the change, the longer it will take to implement. It should evolve naturally over time. A cultural revolution, such as the one experienced in communist China, is too disruptive for people to understand and accept. As a result, they will resist and rebel.

A smaller company can change its culture much more rapidly than a larger company, simply because of communication considerations. In addition, an organization in the private sector can change faster than one in the public sector (such as a government agency), only because a commercial company isn't encumbered with government regulations. This is an instance where a "dictatorship" works more effectively than a "democracy."

To change the corporate culture, one must begin by defining the current corporate and resident cultures, including the customs, religion and society that is observed. There are several indicators for measuring the pulse of the culture: Absenteeism, Tardiness, Turnover, Infractions of Rules, Employee Attitudes, Productivity, etc. All of these can be used to gauge how people behave within the corporate culture.

This is followed by a set of requirements for the culture and a plan to implement them. In a corporate culture, a policy and procedures manual can usually stipulate the customs and society to be observed. Developing a corporate consciousness is far more difficult to implement and involves considerable training and demonstration. Great care must be taken to avoid the "do as I say, not as I do" situation.

It is one thing to enact legislation, quite another to enforce it. Without an effective means to monitor and control the culture, it is quite futile to establish any formal policies or guidelines.

## **SUMMARY**

Management is much more than just meeting deadlines. It is a people-oriented function. If we lived in a perfect world, there wouldn't be a need for managers. People would build things correctly the first time and on schedule, on costs. The fact of the matter is that we live in an imperfect world. People do make mistakes; people do have different perspectives, etc. Management is getting people to do what you want them to do, when you want them to do it. The corporate culture is a vital part of the art of management. Failure to recognize this has led to the demise of several managers. But for those managers who take it into consideration, the corporate culture can greatly influence the productivity of any organization.

**END**

**COPYRIGHT (C) MBA 1995**



**NOTE:** *A modified version of this article appeared in COMPUTERWORLD (August 3, 1992 "CIOs, Get Out of Your Cocoons").*

## ***TODAY'S CHIEF INFORMATION OFFICERS: THE UNTOUCHABLES***

It has often been said that there should be two Presidents for the United States; one to deal with politics, and another to tend to the true affairs of government. The same can be said for today's Chief Information Officers (CIO). Although they should be tending to matters of state, they are all too often preoccupied with politics and gamesmanship.

Ideally, the CIO is the information keystone for a company. As chief architect and information broker, the CIO represents the catalyst between understanding business information needs, and the development organization who must satisfy them. Although the position often comes with much pomp and circumstance, it is all for naught if the CIO cannot effectively tend to this pivotal role.

As the focal point for a company's information resources, the CIO must deal with a wide spectrum of people: end-users concerned with the status of their development projects, as well as reporting problems to existing systems; technicians who argue over tactics of implementation; vendors marketing the latest technical panacea; and CPA's who scrutinize every penny spent by the CIO. Sound hectic? It is. Feeling harassed, the CIO tries to insulate himself, and herein lies the problem.

### **THE ELECTRONIC COCOON**

The CIO begins his tenure as an "ambassador" between his department and the rest of the organization. But as demands close in, he builds a buffer around himself, an electronic cocoon of voice mail and E-mail. Though voice mail is designed to record messages while a person is away from the office, it is primarily used to screen out unwanted callers (both internal and external). Consequently, calls are not returned. E-mail is touted as a convenient way to enhance organizational communications, but the CIO finds himself besieged by a ton of memos and notes (most of which go unanswered). By coordinating these two technologies, it is possible to avoid human contact altogether. However, this would negate the need for the organizational cocoon.

### **THE ORGANIZATIONAL COCOON**

After the electronic cocoon is in place, the CIO develops an "infrastructure" featuring several layers of management. This allows the managers to concentrate on the day-to-day operations of the department, while the CIO concentrates on hobnobbing with the corporate brass. As problems rise through the organization (as they invariably will), the CIO simply adds another layer of management to deal with the problem. In

departmental issues, the CIO is more concerned with who gets to deal with the problem than what the true solution might be.

The CIO's final and crucial sentry is his secretary. Used properly, secretaries play vital roles as expeditors for their managers. For the CIO, the secretary has more of a "pit bull" role, with explicit orders to redirect phone calls and mail, and to tell anyone foolhardy enough to try for a face-to-face encounter that the boss is "in a meeting" and cannot be disturbed.

### **POLITICALLY CORRECT**

The CIO often speaks in a forked-tongue. On the one hand he is conversant in the latest catch phrase (e.g., "global," "infrastructure," "re-engineering," etc.), but on the other he must be politically correct when talking with his peers. Although he balks at technical discussions with his own staff, he loves to overwhelm executive management with his technical verbosity. Conversely, he dazzles the technical staff with management jargon, discussing the "global impact" and "bottom line strategies." As a consequence, the CIO fails miserably as translator between management and the technicians. He plays a different part for each group, making sure neither group can understand (or attack) his grandiose ideas.

### **LOSING TOUCH**

Surrounded by the false security of e-mail and voice mail, protected by platoons of managers and his diligent secretary, the CIO can finally relax. However, due to poor communications with the CIO, executives and users do not know how their business information requirements are being satisfied. And since you cannot communicate with someone who is not there, they become frustrated with the elusive executive. Technicians, awaiting their marching orders, are following a leader who has lost touch with the real world. Impossible to communicate with, he cannot properly manage his department.

Without proper management, chaos will reign, and the CIO's tenure will be brief. Perhaps this is why the average life expectancy for a CIO position is between 6 and 24 months. How can an IS department plan for the future if there is a revolving door at the top? CIO's must shed their insular layers and become accessible to their own people and executives. Only then will information systems be synchronized with the goals of the business.

### **SOME RECOMMENDATIONS**

The CIO is the pivotal player for satisfying the information requirements of an enterprise. The CIO, therefore, must recognize interpersonal communications as an inherent part of the job. Instead of avoiding it, he must master it. Some suggestions:

- 1. RESTRICT THE USE OF ELECTRONIC MAIL** - there are some merits to passing

documents electronically throughout the company. However, legislate the distribution of "junk" mail as a felonious crime.

**2. DUMP THE VOICE MAIL** - its dehumanizing effect is perhaps the biggest irritant around. Instead...

**3. HIRE AN EFFECTIVE SECRETARY** - not just a clerk to chase people away on the phone. A real secretary can expedite problems when the boss is busy or away. The CIO's secretary can be one of the most powerful people in the IS organization.

**4. FLATTEN THE ORGANIZATION** - building an empire with layer upon layer of management only causes confusion in terms of responsibilities and slows the decision making process. Even worse, important decisions tend to fall through the cracks.

**5. TALK IN PLAIN BUSINESS TERMS** - using the latest catch phrases (technical or otherwise) may be trendy but they may also be misleading. Find out what you're talking about, and express it in simple terms. If your executives or technicians cannot follow what you are saying, you are not communicating. If they truly understand what you and your department are doing, you will have real backing and support, instead of a sign-off for the latest superficial offering.

**6. Last but not least, ANSWER THE DAMN PHONE!** People like to know there is a real person out there, not someone who is obscure and runs around the world answering messages by E-mail or voice mail.

END

COPYRIGHT (C) MBA 1995

**NOTE:** *This article first appeared in the ICP Business Software Review, December 1987.*

## ***IN JAPAN "MAYBE YES" MEANS "NO"***

*by Milt Bryce*

In 1976, I made my first trip to Japan. Needless to say, this trip was quite a cultural shock. Here was an environment that was totally alien to me in every way, from food, to language, to social customs. I had been invited to give a presentation (with translator, of course) to a large group of Japanese businessmen on the occasion of my first client's Tenth Anniversary. This company has since become one of the largest providers of programming services in Japan. As in similar situations in the U.S., I decided to begin my presentation with a bit of humor. After being introduced, I walked to the podium, gave the usual greetings to my host and guests and stated, "I feel very much at home here in Tokyo. You have the same cars, television sets, cameras and watches we have in America." I waited for a reaction. Not one of the over 350 businessmen in the audience even chuckled. In fact, there wasn't even a change in expression.

**LESSON:** *"Don't tell jokes in Japan."*

The Japanese are very literal people. What I had said was simply a statement of fact to them and certainly not humorous.

**LESSON:** *"Find out as much as you can about the Japanese and their culture or you will not succeed in their country."*

I entitled this article, "In Japan - 'Maybe Yes' Means 'No'" to illustrate the fact that the Japanese are very polite people. They will avoid being negative, such as saying "No", at all costs. This is an important part of their culture. If you are not aware of this aspect of their culture, you can perceive the wrong messages. For example, many American businessmen have felt they were making effective sales presentations in Japan. Since the Japanese audience nodded their heads up and down during their pitches, they believed the Japanese were buying what they had to sell. When the sales didn't materialize, the Americans were shocked, dismayed and baffled. What they didn't realize was the only thing the Japanese were indicating by their nodding was that they understood what was being said. It did not mean they were ready to buy their products. Making a sale is a completely different process in Japan whether you are selling a product or ideas. They have a procedure called "Ringi" which is very time consuming. It involves everyone having consensus or "Wa." They will not make the buy decision until everyone is in agreement. This process begins at the lowest level in the organization and proceeds upward until it ends with the President. At this point, he asks, "Is everyone in agreement?" If the answer is yes, he applies his stamp or "Ringi" and the decision is made.

I could enumerate ad infinitum the many differences between the Japanese culture and ours. This is not the intent of this article. Rather, I will discuss the Information Systems development environment in Japan and how it differs from our approach in the U.S.

During my first trip to Japan in the mid 70's, I visited several major Japanese companies and toured their systems development departments. It was clear to me they were, at this point in time, considerably behind U.S. companies in the application of computer technology both in hardware and software. In fact, their systems development efforts reminded me of the way we did things in the early fifties in the U.S. when the computer was first introduced. Our systems were mostly batch processes. We programmed in machine code or assembly language. The applications we worked on were conversions from existing tabulating processes or emulations of parts of manual systems. We did very little, if any, real systems work (it also could be said facetiously that we haven't improved much in this respect over time.) The U.S. hardware manufacturers provided the little software that was available as a "freebie." Software packages were virtually unknown and not generally available. This was exactly the same situation I found in Japan. They knew they were behind and were anxious to catch up. As a result, they spent as much time as possible talking with "Gaijin" (foreigners), particularly Americans, in their search for the latest technology.

This brings to mind a popular myth perpetuated about the Japanese, particularly by Americans. You know the one about how they lack "creativity"; *"The Japanese are great copiers, but creative - definitely not!"* Strangely, the Japanese don't mind perpetuating this myth either. The fact of the matter is that the Japanese are very creative and are becoming more so every day. What adds to this misunderstanding is the fact that the Japanese are not "Wheel Reinventors." They consider this a waste of time and money. Something we apparently pride ourselves in doing since we spend a great deal of our time re-inventing based on our "Not Invented Here" syndrome. The Japanese will seek out the best technology available and apply it. In fact, they have institutionalized this process by what they call "Tour Groups." This is where groups of Japanese companies tour America to review what we are doing. For many, this is an annual ritual. This does not seem to affect their egos one bit since they do not suffer from the N.I.H. factor.

**LESSON:** *"Don't underestimate the Japanese."*

Our Automotive, Steel and Electronics Industries did and we are all aware of what happened to them.

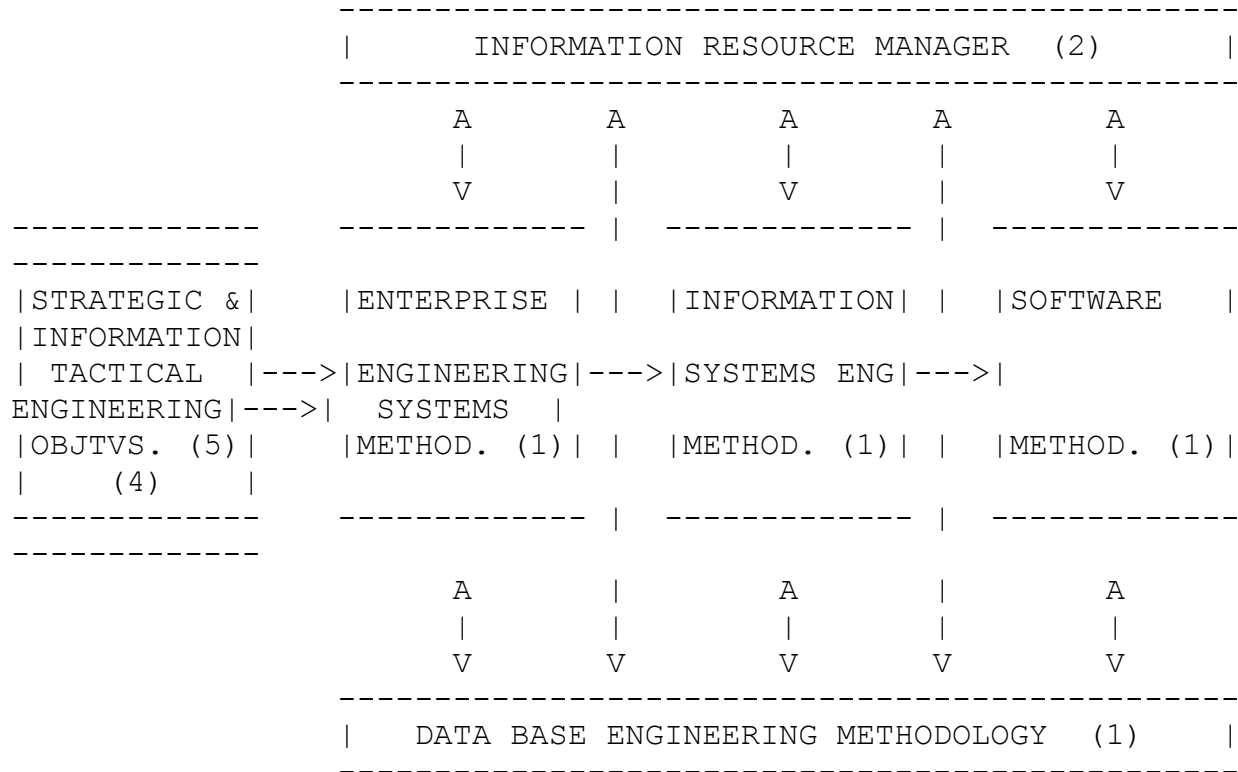
In the ten years since my first visit, I have seen the Japanese progress rapidly. They have followed our developments closely and learned from our mistakes and successes. Added to their successes and failures, they have developed formidable know-how.

You have probably read about the so-called "Software Factories" in Japan, usually

written by people who have not been there. These do not exist as such unless you call what the Japanese computer manufacturers do when they are building operating system software, a "Software Factory." What you will find, however, are "Systems Factories." This is what the Japanese companies have labeled their Information Systems departments. The Japanese, as we all know, are engineering and manufacturing oriented. They organize their Information Systems departments based on engineering and manufacturing principles. They view Information Systems as products that can be built as products using manufacturing/engineering techniques. For example, the methodologies such as Information Systems Engineering, Data Base Engineering, and Software Engineering are considered the engineering and manufacturing processes. Project Control is equivalent to production control. Information Resource Management is the same as the parts department with a Bill of Materials Processor. All of this is intended to produce a quality product (see Figure 1).

FIGURE 1

JAPANESE SYSTEMS FACTORIES



<-----PROJECT MANAGEMENT (3) & QUALITY ASSURANCE----->

- (1) Engineering and Manufacturing Process
- (2) Materials control and Bill of Material Processing (BOMP)
- (3) Production Control and Management
- (4) Product

(5) Enterprise mission and objectives

You will not hear Japanese professionals talking about "Computer Systems" or "Software Systems." They do not believe such things exist. For the same reason they would not call a manufacturing process a "Robot Process." They instead discuss Information, Information Systems and business needs. They realize that the computer and its associated software are only implementation tools for developing parts of systems. They also realize that developing Information Systems is a much larger and more involved process than merely writing computer programs. In manufacturing, robots can perform important tasks but to the Japanese, the manufacturing process is a much more important issue than merely the tools that are used at points in the process. Before they implement a new product, they ask such questions as, "Are we building the correct products?" "Will we build them in the proper sequence?" "Do they meet market needs?", etc. They do the same thing when building Information Systems. Because of this type of thinking and perspective, the Japanese companies recruit new college graduates who have a general academic background as opposed to Computer Science or other such specialties and train them technically at their companies as needed. They want to assure that their people are business oriented as opposed to technically oriented. They believe that systems should serve the information needs of business, and technical needs, although important, are secondary.

The Japanese also have a highly developed sense of order, discipline and planning. This is an integral part of their culture. As opposed to our "ad hoc" and "quick and dirty" inclination to problem solving, the Japanese will develop goals and plans covering long periods of time. Ten years is not considered too long. In the area of "Information Resource Management," they realize it will take a significant period of time to control and document all of their information resources, such as data, systems and people. They, however, are willing to pursue this objective and make the investment since they see the long range benefits. Few American companies will do this or for that matter, see the need. Remember, it is part of our culture to say, "If you are not coding, you're not being productive." Long Range planning in the U.S., may be as long as a year or as short as the end of the day. What I am highlighting in effect is that we tend to respond to the problems of the moment and choose those that give immediate returns, preferably if shown on the Bottom Line in the next Quarterly Stockholders' Report. We are always looking for panaceas and don't recognize there are no "free lunches," only hard work. A current example of this attitude is our fascination with the so-called "CASE" tools. This term is a misnomer in the first place since these tools and techniques do not implement "Software Engineering" in the correct meaning of the term. The looseness of this definition, however, is immaterial but it does illustrate the U.S. approach to systems development and our eternal optimism. We are always looking for magic where none exists. This leaves us wide open for exploitation and inhibits progress. The Japanese on the other hand consider Information Systems development a complex process (hard work) that has to be addressed conscientiously in a thorough and methodical way with a great deal of up-front effort when determining business information needs and requirements.

Another thing you will notice in Japanese "Systems Factories," is a lack of Project Control as we know it. In the U.S. as soon as projects start over-running costs and schedules, the "knee jerk" reaction by M.I.S. management is to install Project Management or more administrative control. They actually believe that this additional overhead will solve the problem. Usually, just the reverse happens and the situation worsens. The real problem is two-fold. One, the department personnel are not trained and do not know what they are doing or how to do it. Secondly, systems personnel are not motivated personally or otherwise. They really don't care whether the job gets done or not. At best, they are mildly interested. This is not a problem in Japan. We chuckle about the idea of "losing face" but this is a real factor with the Japanese. Whenever a project is undertaken in Japan, everyone is involved in the decision. They arrive at a consensus as to "What" will be done, "How" it's done and "When." When the project is started, all members will do whatever is necessary to see that its objectives are met, even if this means working day and night, seven days a week. They don't want to let anyone down or "lose face." In this environment, you don't need Project Control. We lack this motivation and dedication in the U.S. You will see exceptions, of course, but this is rare.

In terms of know-how, amazingly the methodologies being used in Japan for Systems Engineering, Data Base Engineering and Information Resource Management (IRM) are American. This is analogous to the situation with Quality Control. Messrs. Demming and Juran could not sell their ideas to American Industry but found an open mind in Japan. The same with U.S. methodologies. These methodologies are based on engineering and manufacturing and fit the Japanese environment like a glove. The Japanese view tools, aids and techniques such as 4GLs, Structured Programming, Prototyping, etc., as being helpful at the task level, the same as as where a robot performs tasks such as welding, painting, etc. in the factory. For real productivity, an overall framework and structure that formal methodologies provide is required. This is the production line of Systems Development. American Information Systems departments are more tool oriented and lack an appreciation for this broader view. We seem to advocate an "ad hoc" or free form type approach to systems development. As a matter of fact, in my conversations with many American M.I.S. Directors, I am told that to apply a formal methodology would require a cultural change in their organization that they are either unwilling to undertake or find too difficult from a management point of view to implement since it will "stifle creativity". (Another phrase for wheel reinvention).

Language brings a special problem to the Information Systems environment in Japan. Unlike the U.S., where we are concerned fundamentally with only one language or alphabet, the Japanese are concerned with four, Kanji, Katakana, Hiragana, and English. Of the four, Kanji is the most challenging. Kanji consists of at least 40,000 distinct characters and each are unique. What kind of a keyboard do you use to enter them and how do you store them inside the computer or on mass storage devices? Talk about creativity! The Japanese have solved this problem. You now see Kanji word processors, computer printouts, etc. As stated previously, don't underestimate them! I won't bore you with the details of their solution to this problem but I will tell you it takes two bytes to store one Kanji character in computer memory.



On my first visit to Japan, my wife, who is also my business associate, and I visited the main offices of one of Japan's major electronic producers (TV, Stereos, VCR's). During our walk through the offices to the visitor's lounge, where we were to meet with the President and M.I.S. Director for tea and discussion, my wife commented that even though everything seemed similar to a typical large corporation, the ambience was strange. We looked around and lo and behold, there were no typewriters and of course, no noise associated with this standard office machine. This was our first contact with the significance of Kanji in the Information Systems environment. I asked our host about this and he told us that Japanese secretaries take dictation and write all letters in Kanji by hand. These are given to their bosses who in turn sign them with a stamp. They only use a typewriter for letters to foreign clients. This now has changed with the advent of Kanji word processors. It is still quiet, however, since all the machines are electronic.

Lastly, let's discuss that worrisome U.S. problem called "management involvement." If there is anything I have heard over the past thirty years, it's that the U.S. users and management plainly don't want to become involved in Information Systems development. This, however, seems to be changing somewhat, since computer age managers are rising in the U.S. to positions of influence. As far as the Japanese are concerned, this has never been a problem. Japanese executives have a keen appreciation of the value of information. Their superior information allowed them to make significant in-roads in the U.S. and around the world. They realized the need for quality and fuel efficiency a long time before the auto executives in Detroit realized they had a problem. When you attend computer oriented conferences in the U.S., you are inundated "ad nauseum" with propaganda about the latest hardware gimmick or software tools, as if this technology was the only key to our future. In addition, who attends these conferences? You see mostly vendors trying to sell their wares and technicians being rewarded by their bosses with a little R & R. On the other hand, in Japan, just the reverse happens. The conferences in Japan are attended by senior management from Vice Presidents to the Chairman of the Board. The subject matter discussed at these conferences deals with the real substance of the Information Systems field, such as, "How can Information Technology be used to improve the competitiveness of a corporation?" "What is Strategic Systems Planning (SSP)?" "The Role of Information Resource Management in the future", etc. Japanese management is information oriented and they leave the details of technology to the technicians after they have specified the What, Why, and When of the problem and given direction. They are firmly in control and consider information systems the "life blood" of their companies.

In the foregoing, I have outlined some of the major differences between the U.S. and the Japanese Information Systems environments. In answer to the question "Are we ahead or behind?", it depends on what areas you are talking about. In the area of Information Systems development and management, we are falling rapidly behind. In the area of computer technology and software, we are still ahead but only barely. As Pogo said, *"We have met the enemy and it is us."* We are so imbued with the way

we view things and so arrogant about the righteousness of our ways, we never seem to find the time or the initiative to search out obviously effective ways of accomplishing goals that come from a different culture. We must have the open-mindedness to entertain the possibility that other cultures may have a wisdom or an insight that our own narrow viewpoint obscures. This may be the ultimate lesson we learn from Japan.

Are the Japanese behind? ***"Maybe Yes."***

END

COPYRIGHT (C) MBA 1995

**NOTE:** *This article first appeared in the October 1991 issue of MBA's Management Visions newsletter.*

## ***THE LADY OR THE TIGER?***

In the short story, 'The Lady or the Tiger?,' a young man is forced to make a difficult decision. Behind one door is the maiden he hopes to wed, behind another door is a ferocious tiger and certain doom. The problem lies in the fact that both doors look alike and there is no way to differentiate between the two. The lesson of the story is simple: during our lives we are forced to make some very difficult decisions based on insufficient information.

This same scenario is being played out in countless IS organizations around the world. Under pressure to produce information systems, IS management is searching desperately for a panacea for their woes. A variety of vendors and consultants have responded with a diverse set of remedies, not all of which are compatible or consistent. Even though the jargon of these products sounds the same, they have entirely different meanings and objectives. Such inconsistencies make it difficult for consumers to differentiate between products and integrate them into their development practices. For example, ask two CASE vendors (Computer Aided Software Engineering) to define their terms and you will probably come up with two distinctly separate lists.

Such inconsistencies are indicative of the state of the industry. It would be safe to assume that we have not yet attained 'profession' status in the area of managing information resources. Computer technology is one thing; Information Resource Management is another. While the 'experts' argue over information theory, the western world is rapidly losing its edge; we refer to this as the 'rearranging the deck chairs on the Titanic' phenomenon.

The design, development and management of information resources does not have to be based on cryptic concepts and terminology. In fact, it is a teachable science with simple governing principles and rules. It has always been our contention that the best solutions have been those that were simple and based on common sense.

### **MANAGEMENT ISSUES**

It is almost unanimous among authorities in the field that the best way to implement application development aids such as CASE tools is to first establish a strong management framework by which these tools can be orchestrated. In manufacturing, this type of activity is referred to as 'Industrial Engineering,' a discipline aimed at establishing the assembly lines of the business. As workstations are established and the flow of work is defined, the Industrial Engineer considers where various techniques and tools are deployed and synchronized with the assembly lines. The Industrial Engineer constantly monitors the assembly line and introduces new and improved

techniques and tools to replace old ones. What this highlights is simple: without defined assembly lines, tools and techniques will be misapplied to the point of being counter-productive.

This concept of Industrial Engineering is implemented in MBA's "PRIDE" INFORMATION FACTORY, a product that simulates a factory-like environment for Information Resource Management. The product defines three complementary assembly lines (methodologies) for designing and manufacturing information resources:

- \* Enterprise Engineering Methodology (EEM) - a phased approach for defining the enterprise and formulating an enterprise information strategy synchronized with business objectives.
- \* Information Systems Engineering Methodology (ISEM) - a phased approach for designing and manufacturing total information systems, not just the software portions. Software Engineering is considered a subset of ISEM.
- \* Data Base Engineering Methodology (DBEM) - a phased approach for designing and developing the corporate data base. The intent is to build data base synchronized with the information systems of the enterprise.

The INFORMATION FACTORY offers many of its own tools and techniques to be used along the assembly lines, but they are not all inclusive. Because of this, MBA recognizes that customers will want to use a variety of additional techniques and tools, particularly in the area of software engineering and physical data base design. Consequently, the INFORMATION FACTORY offers an open architecture thus allowing additional tools to plug in and out of the environment, thereby integrating the development process.

A totally integrated development environment does not happen by chance. It requires considerable planning and hard work (e.g., the Industrial Engineering analogy). But the real catch is IT REQUIRES MANAGEMENT.

A defined and integrated development environment, as described by 'PRIDE,' represents organization, discipline, cooperation, and accountability. This goes totally against the grain in most data processing organizations. Consequently, the adoption of a management methodology is avoided in favor of CASE tools that do not require management commitment (the tool either works or it does not; another one can be easily substituted).

All of this represents an interesting dichotomy: people are beginning to understand that CASE alone is not the answer for solving their development problems (that strong management is actually the right answer), yet, they are unwilling to do what is necessary to make it work. Instead, IS management opts for a 'quick and dirty' solution that will seemingly solve the problem on a short-term basis (until they can move on to their next job).

Unfortunately, companies are finding out the hard way that these short-term solutions present some long-term headaches; e.g., lack of integration between tools, redundancy of effort, applications do not share data, no continuity between development projects, longer development cycles, etc. By ignoring the problem, most IS executives hope that it will simply go away. However, it will inevitably return with a vengeance.

Even when forced into acquiring a methodology, IS management will typically take the most non-committal and superficial approach possible, thereby becoming nothing more than 'window dressing' to impress an auditor. It is rather ironic that companies are willing to spend millions of dollars on computer technology, yet balk at acquiring management technology; a sort of 'penny-wise, pound-foolish' behavior.

Ignoring such serious management problems is tantamount to corporate negligence. It will only delay or encumber the company's ability to respond effectively to competition.

### **WHAT CAN BE DONE?**

The "PRIDE" INFORMATION FACTORY alone is not the answer. Unless a change of corporate culture is effected, even "PRIDE" is destined for failure. It is time for corporations to do some soul searching and reflect on their corporate policies and attitudes towards managing information resources. There are many things that can be done to alter the corporate culture. The following list represents the basics:

#### **1. CHANGE INCENTIVE PROGRAMS**

Most western companies compensate employees individually on a week-to-week or month-to-month basis, regardless if anything is produced or not. Consequently:

- \* The individual thinks no further than the next paycheck.
- \* The emphasis is on time spent as opposed to accomplishments achieved.
- \* Individualism is promoted as opposed to teamwork and cooperation. Competitive politics emerge ('one-upmanship').

Instead, incentive programs should be changed to manipulate personal attitudes. Compensation should take into account:

- \* Goals attained over a long period of time (e.g., one year, five years, ten years). This promotes employee longevity and raises the consciousness of corporate long-term goals.
- \* Reward based on group effort, not individual effort. This will force people to work together toward common goals and promote cooperation. It could even create an esprit de corps.

\* Compensate based on deliverables produced, not hours worked. This will force employees to concentrate less on watching the clock and more on producing products. The only problem here is that if deliverables are not precisely defined, it would be impossible to measure how well they were prepared. Consequently, sloppy work could be produced. This gives rise to the next point...

## **2. CHANGE THE PERSPECTIVE IN DEVELOPMENT FROM AN ART TO A SCIENCE**

If development is regarded as nothing more than an elegant art form, where developers are treated as 'free-spirits' who cannot be encumbered by organization and discipline, then chaos will reign. Just as the Berlin Wall inevitably was broken down, the 'black box' mystique to systems development must be taken apart brick-by-brick.

To make this happen, several things have to occur:

\* A Chief Information Officer (CIO) or Information Resource Manager (IRM) has to be appointed who will act as the principal information architect or information broker for the enterprise. It is essential that this person be a businessman as opposed to a technician (ideally, this person should NOT come from computing).

\* Define the governing concepts and philosophies for design and management. This should address resource management, project management, quality assurance, engineering and manufacturing. Avoid cryptic and unproven theories; keep it simple. Use common sense principles that work. These will be easier to communicate and uphold over time. Also, define your terms, thereby establishing the vocabulary for design and management.

\* Formally define the work environment for implementing the governing concepts and philosophies. This is the area of methodologies which defines WHO, is to perform WHAT, WHEN, WHERE, and WHY (the 5-W's). Ideally, the environment should focus on design correctness (proof that a design satisfies specifications), and the production of a quality product. It should also specify the duties and responsibilities of the people in the work environment, benchmarks and deliverables, and review points. The intent is to define an environment where results can be monitored and measured.

\* Define the tools and techniques to be deployed throughout the workplace. Such tools and techniques should not run contrary to the governing principles and defined work environment. They must be able to integrate within the methodologies with little or no difficulty.

\* Move away from job titles such as 'programmer' and 'analyst,' and embrace titles such as 'engineer' and 'architect' instead. Although such a change is actually cosmetic in nature, it tends to change the mind-set of people to think and act more as professionals. (Even the initials 'MIS' or 'DP' should be dropped in favor of a more

credible organization; such as 'IRM').

This is the category where the "PRIDE" INFORMATION FACTORY can offer the most support. As a management philosophy, it is used to define the work environment along with the duties and responsibilities of the people in the workplace. Its approach to management and design is based on common sense principles that are universally applicable.

### **3. WEAVE IRM INTO THE MANAGEMENT FABRIC OF THE ENTERPRISE**

The lack of credibility has perhaps been the biggest stigma associated with IS organizations. This is primarily because of the technical mumbo-jumbo and poor performance normally associated with data processing. Consequently, IS executives are not included in the mainstream of business planning at the executive level. To overcome this problem the IS organization must demonstrate that it has reformed its mode of operation. This means educating executives in the governing principles of information resource management and the revised methods of operation.

Assuming IS has proven its credibility, the CIO/IRM must become an integral part of executive management. Such participation will put the IS organization in tune with corporate direction, thereby becoming more effective in terms of serving the information needs of the enterprise.

Executive management must also participate in the development of a corporate-wide information strategy. Such a strategy is not so much concerned with information related technology, as it is towards establishing the direction of development projects in support of business needs. Such an Enterprise Information Strategy (EIS) establishes the priorities of the business, thereby moving the company as a whole towards common objectives.

### **4. LAST, BUT NOT LEAST - MANAGE!**

This is perhaps the biggest weakness in the whole formula for change. All of the available management tools on the marketplace will not make anyone a better manager. The tools can provide some important information, but unless the human being can effectively follow through on the information, they are for naught. This highlights the fact that management is a people-related issue, not an administrative issue. Management means leadership, communications, organization, motivation, delegating responsibilities, discipline, and holding people accountable for their actions. Management represents the 'Achilles' Heel' of most IS/DP organizations. Just because someone is proficient in a certain technology does not qualify them in management practices. These are skills that have to be taught and practiced. People are not born with management skills. They must be groomed in the science of management instead. It is hard work requiring specific talents.

As one indicator of the problems in IS management, consider the trend towards

"outsourcing," where an internal IS department is replaced by a contracting company to run IS. "Outsourcing" is an admission that companies do not know how to manage IS. What they do not realize is that the outside contractor is probably just as inept as the internal employees they are replacing.

## **CORPORATE CULTURE**

Changing the status quo is not a simple task. People have a natural aversion to change (particularly in IS/DP). Platitudinous statements will not suffice. Visible changes must occur to demonstrate the corporation's commitment to change. After all, it is one thing to legislate a law, quite another to enforce it.

It must be remembered that culture is learned. As such, it can be taught and enforced. Changing the culture involves influencing the company's customs, philosophy, and society. Therefore, the impetus must come from executive management. If they do not realize the severity of the problem, they will never commit to such cultural change.

This is the edge the Japanese have over the western world. They have a keen awareness of the role of corporate culture and go to great lengths to manipulate it. Being a homogeneous society to begin with, they are not plagued by the heterogeneous people problems presented to their western counterparts. Consequently, their incentive programs, work environments, and management philosophies are more in tune with those mentioned in this article. Are the Japanese smarter? Not necessarily. Many of the management concepts used in Japan were actually introduced by Americans (a la Deming, Juran, Kepner-Tregoe, etc.).

To deny that such problems exist in the IS/DP world, is to deny reality. A 'wait and see' attitude will only postpone the inevitable. This was the attitude American automotive manufacturers took in the 1970's. They are still feeling the consequences from a delayed decision to change.

The problems in IS/DP are real. The patient is sick. And the problems will not go away with superficial remedies. In fact, the cure will be quite painful. Hard decisions have to be made: Do we delay the inevitable and continue to conduct business as usual? Or do we address the problem head-on? The question is simple: The Lady or The Tiger?

**END**

**COPYRIGHT (C) MBA 1995**



# ***BACK TO THE FUTURE: RE-INVENTING RE-ENGINEERING***

*July 13, 1992*

*"The word "re-engineering" implies something was "engineered" in the first place, which is rarely the case."*

*- Milt Bryce*

*"The common retort is, "Why replace these applications? They may be old, but they're running!" The answer is: They are costing companies more than they think."*

*- Ralph Sprague, Barbara McNurlin (1)*

Controlling the elements of an information system along with the process by which they are assembled are critical to success in development. Companies are now beginning to realize it is no longer a matter of how fast software is written; instead, building the "right" software is far more important. This is what has led to the "Re-Engineering" movement in the IS industry. The lackluster results produced by Computer Aided Software Engineering (CASE) tools have caused many companies to challenge not only what software is produced, but the business processes to be supported. Re-engineering could be a viable solution, if the industry can ever define it.

Unfortunately, re-engineering suffers from the same Madison Avenue hype as other panaceas (e.g., CASE). It promises to increase customer satisfaction, improve quality, reduce turn-around time, increase staff morale and slash costs, all at the same time. At least that's what the seminars and advertisements promise. Of course, the "placebo of choice" of the 80's, CASE, made similar promises. And in spite of the nifty diagramming techniques, rhombuses and 4th generation languages, the tools could not live up to the hype. However, CASE tools DO work - but their forte is limited to software engineering, not systems engineering or business planning, which are sorely needed if you want to develop the right software.

So what is re-engineering? There are different types addressing different things. "Software Re-Engineering" focuses on maintaining computer software only. But what is the point of maintaining a program if it no longer serves a business purpose? Herein lies the problem and the need for what is today called "Business Re-Engineering." It is this latter form of re-engineering that is in vogue today. But without a clear delineation between the two, both efforts will produce incompatible results.

There are four basic objectives to "Business Re-Engineering":

- 1. DEFINE THE BUSINESS** - an identification of the business functions implemented by the enterprise. This establishes the mission of the business and the scope of its operations.

**2. DEFINE INFORMATION REQUIREMENTS** - an identification of the intelligence required to manage/operate the business. Information requirements support the actions and decisions of the business as specified by the business functions.

**3. DEFINE BUSINESS PROCESSES** - a design of the overall processing components required to support the information requirements of the business.

**4. IDENTIFY THE NEED FOR SOFTWARE** - a determination of the computer programs needed to support the business processes.

There are, of course, other matters for consideration in "Business Re-Engineering," but these represent the primary concerns.

## **EFFECTIVENESS VERSUS EFFICIENCY**

All of this, of course, relates to productivity; but to truly appreciate the need for "Business Re-Engineering," we must first understand the differences between efficiency and effectiveness, and how they relate to productivity. Bryce maintains that the concepts of effectiveness and efficiency are erroneously viewed as synonymous.(2) In reality, they are separate and unique entities. Efficiency addresses HOW an operation is performed (e.g., speed of development); Effectiveness addresses WHAT is to be performed. In other words, are we doing the "right" things (effectiveness), and are we doing things "right" (efficiency)? This leads to the Bryce formula:

$$\text{PRODUCTIVITY} = \text{EFFECTIVENESS} \times \text{EFFICIENCY}$$

Bryce argues, *"There is nothing more unproductive than to build something efficiently that should never have been built in the first place."* Application development aids, such as CASE tools, address the efficiency issue. "Business Re-Engineering" addresses the effectiveness issue.

## **RE-ENGINEERING VERSUS "METHODS & PROCEDURES"**

*"To start our consideration (of the management system) and how to improve it, let us ask three questions:*

- 1. Does anyone know exactly what information each man and each woman in the organization needs to carry on work?*
- 2. How much of the effort now expended on computerization makes a direct contribution to essential systems actions? And, conversely, how much work in the computer room is necessary just because you're using a computer?*
- 3. If you can examine a system, how can you judge if it is a poor system or a good one?"*

- Leslie H. Matthies

(3)

True re-engineering isn't new; we've been asking these questions for years. In fact, Matthies was assisting companies in defining business procedures and their requirements long before the common business processes became automated (see chart). However, when these processes became automated, people falsely assumed computers would naturally organize the procedures. True "business analysts" were traded in for technicians who specialized in software design and development. Regrettably, "Methods & Procedures" departments became a thing of the past.

What did the "Methods and Procedures" departments actually do? They would analyze the operations of a company; this included basic systems work, forms design, work simplification, and work measurement to determine what the company and its people were doing. They determined which procedures were efficient, which were effective, and which ones needed to be changed. In addition, they also evaluated the use of office equipment (e.g., tabulating equipment, etc.) Sound familiar? The "new" discipline of "Business Re-Engineering" covers the same concerns.

Systems Analysts too often concentrate solely on the complexity of software. The computer has not solved the problem of process organization, nor has the software. So now, with multiple systems, hardware, software and backlogs, we are re-examining what the company is doing. This is one of the key reasons why re-engineering is popular today. Re-examination is good, but it is not the equivalent to re-engineering.

## **TIME FRAMES**

Evaluating the use of software will not help in the formulation of a corporate information strategy. To illustrate, it will not provide any insight into the organization and operation of a business. The processes themselves and their specifications must be examined. When software is examined, process timing information is not defined; for example, there should be specifications for when a process occurs and how long it takes to perform it.

Specifications must originate from information requirements. Information supports the actions and decisions of the business; as these actions and decisions exist within a unique time frame, information is time dependent. Therefore, the business process responsible for delivering the information must also exist within a unique time frame.

In the discipline of systems process management, time and motion studies are a key part of any systems study. Such studies help determine the "best" way to accomplish a task (motion study), and "how much time" it takes to accomplish the task (time study). This helps to determine the proper tasks for the process, and eliminates errors.

Bryce clarifies information timing: *Information is a perishable commodity. It only has value within a specific point in time. It is not sufficient to merely determine what data is necessary; it is also necessary to define the timing of the information.* This

timing will ultimately dictate how data will be collected, stored, and retrieved to produce information." Bryce says all business processes operate within time frames, such as instantaneous, daily, weekly, monthly, quarterly, etc. And suggests making use of time considerations during design, rather than correcting the system later.

Timing can be a good measure of how the current system is working; what works, what doesn't, and what can be re-used. Timing is an important determinant for effective business process re-engineering.

## **ACROSS-THE-BOARD ANALYSIS**

Information and business processes cross organizational and departmental boundaries. To effectively implement re-engineering, we need to examine our business and its practices as a whole, not one department. We must be able to accurately define:

- \* The scope of the business.
- \* How various business functions interrelate/communicate through common systems and data.
- \* How the business is physically organized. This includes an organizational analysis to denote overlaps in responsibilities and omissions in fulfilling business functions.
- \* How well existing systems support the information requirements of the business. This includes a description of where the company is well supported, and where it is weak.

A true analysis will answer the questions WHO/WHAT/WHERE/WHEN and WHY (the five-W's).

When performing such analysis, some companies are sidetracked by concentrating solely on the data aspects of the business. Although this is helpful, it doesn't answer the more important question regarding information utilization; for example, "When the information is delivered, what will the user do with it?" In other words, what business actions and decisions are supported? Considering a company's data requirements is important, but should fall outside the realm of "Business Re-Engineering." Too often companies become immersed in data base studies that have lost sight of the main objective. Consequently, the enormous data models created by such effort lack applicability to the company's information requirements. At most, business analysis should consider the major facts and events (objects) affecting the business. There is a time and place for everything. Creating detailed data models during a business study is not one.

## **ENTERPRISE ENGINEERING**

Perhaps a more appropriate name for "Business Re-Engineering" is Enterprise

Engineering; however, it also has different definitions. Bryce views Enterprise Engineering as the discipline of defining business (organizational) resources and the development of an Enterprise Information Strategy synchronized with the business. Sometimes, Enterprise Engineering is confused with the activities associated with Data Base Engineering. Data Base Engineering is more appropriate for data base engineers as opposed to business analysts.

The objective of true Enterprise Engineering is to:

- \* Define/model the business (both the logically business functions and physically organization).
- \* Make recommendations for changing the physical organization to more effectively implement the logical.
- \* Develop an Enterprise Information Strategy based the goals of the business (e.g., strategic and tactical).

Enterprise Engineering is a precursor to all other efforts, such as Information Systems Engineering and Data Base Engineering. It determines the essential requirements necessary to run the business. The end result is an Enterprise Information Strategy; a "road map" for the company, defining the information priorities of the business. Such insight is necessary in order to effectively deploy development resources; after all, you should never embark on a journey if you do not know your final destination.

The Enterprise Information Strategy will not be a static plan; it will be dynamic, changing with the business. A static plan cannot adapt to constantly changing business conditions (e.g., economics, politics, competition, government regulations, etc.). Therefore, the strategy must be organized to accommodate change.

### **WHO SHOULD PERFORM "BUSINESS RE-ENGINEERING"?**

Ideally, a Business Analyst or someone from the user community should be charged with this responsibility. Unfortunately, "Business Re-Engineering" defaults to programmers who are not necessarily the best suited to perform the role. Such activity requires someone who is more in tune with the business as opposed to technology. After all, "Business Re-Engineering" addresses management issues, not computer issues.

Implementing "Business Re-Engineering" often requires a change in the organization of development resources (see chart). Instead of the proliferation of programmers, it is necessary to delineate the duties and responsibilities of separate development functions; e.g., Enterprise Engineers, System Engineers, Software Engineers, Data Engineers, DBA's, etc. Each is charged with developing different types of information resources. For example,

- \* Enterprise Engineers develop organizational resources (business functions, organizational entities, information requirements, business objectives, projects, etc.).
- \* Systems Engineers develop the logical system resources (systems and business processes).
- \* Software Engineers develop physical system resources (programs, modules, subroutines).
- \* Data Engineers develop logical data structures.
- \* DBA's develop physical data structures.

Although each development function is concerned with different types of information resources, they can all be integrated and shared, thereby improving communications between developers. An organized and managed development environment offers control and the confidence that you are moving in the right direction. And that's what "Business Re-Engineering" is all about - helping your organization better manage its information resources, today and tomorrow.

#### REFERENCES:

1. "INFORMATION SYSTEMS MANAGEMENT IN PRACTICE" - Ralph H. Sprague Jr., Barbara McNurlin, 1986, Prentice-Hall, ISBN 0-13-464934-6
2. "THE IRM REVOLUTION: BLUEPRINT FOR THE 21ST CENTURY" - Milt & Tim Bryce, 1988, MBA Press, ISBN 0-9621189-0-7
3. "THE MANAGEMENT SYSTEM, SYSTEMS ARE FOR PEOPLE" - Leslie H. Matthies, 1976, John Wiley & Sons, ISBN 0-471-57697-2

END

COPYRIGHT (C) MBA 1995

**NOTE:** *This article first appeared in the April 1991 issue of MBA's Management Visions newsletter.*

## ***AN INTERVIEW WITH MILT BRYCE***

*Twenty years ago this month, Milt Bryce decided to go for broke. Reflecting on his future he did not believe he wanted to spend his time correcting, rebuilding and staffing IS organizations. He believed he could package his know-how and help organizations prepare for the future. With this goal in mind, he resigned from his position, bought some basic office equipment, and along with his wife Jean, set up an office in the back room of his house in Cincinnati. One of his first steps in starting his company was to take out a large sheet of butcher-block paper and using a template drew out the phases, decision points, deliverables and processing flow of what was to become the world's first commercial system design methodology, "PRIDE" (P<sub>R</sub>ofitable Information by D<sub>E</sub>sign). The rest, as they say, is history. Since it was introduced in 1971, "PRIDE" has grown and evolved into a robust product line with customers all over the world.*

*To celebrate our twentieth anniversary we decided to interview the creator of "PRIDE", Milt Bryce, MBA's President & C.E.O. His outspoken comments and observations about the industry are just as insightful today as they were back in 1971.*

**Q:** WHEN YOU WERE ORIGINALLY SETTING UP THE COMPANY, WHAT DID YOU HOPE TO ACCOMPLISH?

**BRYCE:** Based on my experience as an MIS Director, I felt that it was absolutely essential that I establish a company that would assist other companies in the development of systems. In other words, developing the standards, procedures and methods for managing an information systems organization. As a director, it became blatantly obvious to me that such a service did not exist. People were operating on an 'ad hoc' basis. I felt that I could become the standards organization for companies and the industry in general. This has happened with many companies. We are looked upon as their standards department. This makes a lot of sense because most companies are not in this business and cannot afford to spend the time to develop their methodologies and their management approaches. So, they use us in this capacity.

**Q:** SO YOU SEE THE COMPANY MORE AS A MANAGEMENT CONSULTING FIRM AS OPPOSED TO A SOFTWARE VENDOR?

**BRYCE:** We only produce software to implement our concepts and philosophies. We do not consider ourselves a software vendor as such. We consider ourselves management consultants specializing in the area of Information Resource Management. By the way, one of the areas we are moving strongly into is providing methods, techniques and tools for the Systems Integration business. We see this as a large

marketplace in the future. One of the things wrong with Systems Integration today is that these companies lack the concepts and tools needed to drive this type of business. We think of the "PRIDE" Information Factory as providing the 'engine' for the Systems Integration field. Some people in this area have already recognized this and we're developing partnerships with them. We'll provide the 'engine' and they'll provide the manpower to build the applications.

**Q:** WHAT WERE THE EVENTS THAT LED TO THE ADVENT OF "PRIDE" BACK IN 1971? IN OTHER WORDS, WHY "PRIDE"?

**BRYCE:** The creation of "PRIDE" was based on my experience as an MIS Director for two major corporations. After my tour of duty at UNIVAC, I went to work for the Quaker Oats Company and, subsequently, with the U.S. Shoe Corporation. At both of these companies I found the MIS function totally disorganized. They had no methods, procedures or anything. It occurred to me that this was probably true for many companies in the United States. Therefore, I decided to leave my job as MIS Director for U.S. Shoe and created the "PRIDE" methodology. By the way, this was the first commercial 'methodology'. There were already many approaches for project management but I wanted to address the problem from a DESIGN point of view, with project management and documentation as by-products.

**Q:** WAS "PRIDE" HARD TO EXPLAIN AND SELL IN THE EARLY DAYS?

**BRYCE:** It still is. Back in the early days when I was 'cold calling' on the telephone I had a difficult time trying to describe the product. MIS Directors would say, 'Well what exactly are you trying to sell?' I had trouble finding the words at first but then said 'methods'. They would ask, 'Methods for doing what?' 'Methods for designing systems; a Systems Design Methodology,' I replied. And that's where the term 'SDM' came from. But it was extremely difficult since it was such a new concept. People understood the need for standards for documentation and project management but they did not understand the concept of a formal methodology for designing systems.

**Q:** HOW HAS "PRIDE" EVOLVED SINCE ITS INTRODUCTION?

**BRYCE:** When we introduced "PRIDE" in 1971, it set many precedents for the industry. Unfortunately, many of these precedents were not understood. For example, it was the first methodology aimed at the design of information systems. But at the time we introduced it, people were only interested in project management and documentation. What MIS Directors didn't realize was that you can't manage projects unless you have an organized methodology in place first. If the methodology is design driven, documentation is a natural by-product.

"PRIDE" also introduced the concept of Data Management and the Data Dictionary. Even though the dictionary was implemented manually, it still set the precedent of providing a vehicle for controlling data and its relationship to process components.



Data Management is very important. This is a concept I introduced in 1965 at the Quaker Oats Company, along with a primitive data dictionary. This was a necessity since we were trying to produce a complete MIS system. You cannot possibly design integrated systems unless you manage the data. People at the time didn't realize this. Back then, the industry was mostly developing batch systems, using flat files. The concept of Data Management, at that time, didn't seem important. But data is the glue that holds systems together. In order to produce information, you must first control your data and processes. The concept is really quite simple: INFORMATION = DATA + PROCESSING. In other words, you have to control both data resources and your processing resources. This is the rationale for an Information Resource Manager (IRM) as opposed to a simplistic data dictionary.

One of the senseless arguments in our industry is whether you design systems based on a 'data-driven' or 'process-driven' approach. The answer, of course, is both. But that's not the point. You shouldn't be only thinking of data or processing; you should be thinking 'information'.

"PRIDE" went on to automate the Data Management concept in 1974 and we were copied by many people, including IBM and others. But none of them truly understood the concept. Due to their DBMS orientation, they were only concerned with data resources, and not the other resources, such as processing and business resources. This is why we switched the name of our LOGIK Dictionary to the Information Resource Manager (IRM) in 1981, to distance our product from the other dictionaries on the market. In 1979 we were also able to interface the IRM to various Data Base Management Systems (DBMS) and program generators, particularly those that were specification driven. This is a big deal now, but keep in mind, we were doing this in 1979. Most people didn't realize what we were doing. It was simply a too advanced concept for most people to understand. Here we are in 1991 and the CASE people are just starting to try to do what we did in 1979.

Our product evolved into three complementary methodologies, one for Enterprise Engineering, one for Information Systems Engineering, and another for Data Base Engineering. By the way, one of the most ridiculous concepts I have ever heard is 'Information Engineering'. How do you engineer information? All you can really engineer are the components used to produce information, such as data and processing. I also believe that the gurus that talk about 'Information Engineering' know nothing about 'information' or 'engineering'. Bear in mind, in this industry, you don't have to be logical or intellectually correct to sell an idea, just use a lot of marketing hype. Shoot the baloney and people will buy it. Unfortunately, this is what's behind a lot of the CASE tools; they're based on very shallow thinking.

At present, we have created the concept of the 'Information Factory' and I'm waiting to see what the copy-cats will do now; whether they'll write books from our sales brochures or whatever.

Other innovations I am particularly proud of is the introduction of the Information

Requirement component - a complete specification for information. We're the only ones that I know of who have produced a tangible specification for information; everyone else dances around the issue. Also, our Data Base Engineering Methodology (DBEM) introduced the concept of 'objects', a major advancement over the entity-relationship model. As it turns out, our 'object' concept happens to be compatible with others in the field called 'object oriented programming' and 'object oriented DBMS'.

**Q:** LET'S FOLLOW THROUGH ON SOMETHING YOU JUST MENTIONED IN TERMS OF DATA MANAGEMENT. WHAT IS WRONG WITH THE INDUSTRY'S PERCEPTION OF THE DATA DICTIONARY?

**BRYCE:** The Data Dictionary as perceived by the industry today is nothing more than a programmer's tool. It's a cryptic approach for defining data resources, nothing more. The whole idea of defining the business and system resources is foreign to the technicians. Their interest is simply not there for obvious reasons. However, we see the data dictionary as a 'Bill Of Material Processor' (BOMP) as used in manufacturing, which, although is a simple concept, is too sophisticated for the software people to assimilate. This is why we renamed our dictionary to the IRM, a much more global perspective to the resource management issue.

**Q:** HAS STAYING THAT FAR AHEAD PRESENTED A MARKETING PROBLEM TO YOU?

**BRYCE:** Being ahead in a field such as ours does present a problem. People are not focusing on the solutions you are thinking of. They're only thinking of current solutions. Finding 'forward thinking' people is difficult. In fact, I have heard it said many times that our products do not fit what our competition offers. Thank God for that. We're way ahead. Things like CASE tools offer some razzle-dazzle for developing programs, but they cannot help in an overall Information Resource Management strategy. This is where we're concentrating.

**Q:** HAS THE CLIMATE FOR "PRIDE" CHANGED SINCE 1971?

**BRYCE:** I don't think the climate has really changed. It's still pretty much the same. People are still way behind. The concepts embedded in "PRIDE" and our Information Factory are still ten to fifteen years ahead of the industry.

In the United States most managers are doing nothing more than attacking symptoms, which is no different than they were doing twenty years ago. The industry is still programming oriented. It's kind of ridiculous. It's like on an assembly line in a manufacturing plant; we've got welding robotics on the line, but nobody's challenging whether the product we're welding is any good or not. This is the same thing in the information systems field; we have all of these technical tools and cute programming aids but fundamentally we're not addressing the real problems of defining information and building the right systems to produce it. And that hasn't changed in the last twenty years.

Now I do see a lot of this changing rapidly over in Japan. Everybody thinks the Japanese are technically way behind us. I'm sure there is a certain amount of truth to that in the area of software but I tend to believe that they will let the U.S. attack the software engineering problem while they attack the systems engineering problem, which is a much more encompassing problem. You know, even if the system is physically implemented somewhat poorly, yet solves the business problem correctly, then you will be successful. But an elegantly programmed system that solves the wrong problem accomplishes nothing.

**Q:** "PRIDE" IS SOMETIMES REFERRED TO AS 'THE BEST KEPT SECRET IN THE WORLD'; WHY DO YOU THINK THAT IS?

**BRYCE:** I'm not sure. I guess, it's like comparing gold with lead. If you don't know the difference between the two, lead will function just as good as gold. I think you will find that's true in our industry. People do not want to think in an engineering/manufacturing way. They're more concerned with technical elegance and facade as opposed to substance. They've forgotten the purpose of our field: to solve business problems. But they're still worrying about the technology as opposed to anything else. In this type of situation, "PRIDE" will always be a secret since it addresses business issues as opposed to technical detail.

Up to now, we've kept a low profile as we were developing the Information Factory. But you're going to see this change when the product comes out.

**Q:** HOW HAS THE COMPETITION CHANGED OVER THE YEARS?

**BRYCE:** This is kind of interesting. Initially, our competitors were companies that stressed documentation or project management. For instance, SDM/70 was originally called 'Segmented Documentation Methodology' which was a very apt name for the product. From the accounting firms, came numerous paper driven methodologies aimed at managing projects using the classical 'waterfall' project management approach.

You have to be careful with the expression 'SDM'. There's a big difference between a Systems DESIGN Methodology, such as "PRIDE", and a Systems DEVELOPMENT Methodology. The intent of a design methodology is to do just that, design. How you manage the design effort is independent. It's just like in architecture where there are phases by which a product, such as a building or a house, is decomposed into levels of detail, from the abstract to the specific. Project Management can be added to monitor and control this effort, but this should not be confused as a part of the design issues. On the other hand, Systems DEVELOPMENT is a clever way of saying that you wish to manage the development process. This means a project management orientation; not design. It doesn't give you the governing rules and procedures for design.

A methodology should address the five W's (WHO is to do WHAT, WHEN, WHERE, and

WHY). This is important since it defines your working environment. Other people concentrate on techniques and tools, which address HOW in the methodology. Most people are sloppy when discussing this subject.

**Q:** SO NOT ALL METHODOLOGIES ARE CREATED EQUALLY?

**BRYCE:** No they're not. As a matter of fact, most of the methodologies that people are talking about, such as 'Structured Analysis' and 'Information Engineering' are not really methodologies at all. They're techniques. And they're limited techniques. Most of the CASE products are nothing more than tools that implement these techniques. I've heard many of these things being called 'shelfware' because people do not know how to put them in a methodology environment. By the way, don't get me wrong, there are many CASE tools out there that are effective in doing certain activities, within a methodology. But if they don't have a framework from which to operate they'll become useless and end up on the shelf. I think we have to first have an overall framework under which to build systems.

**Q:** SO THEN YOU SEE METHODOLOGIES FROM A MANUFACTURING PERSPECTIVE IN TERMS OF DEFINING THE ENVIRONMENT?

**BRYCE:** Yes, in the sense that it resembles an assembly line. However, it encompasses both engineering and manufacturing. For example, the engineering aspect is when you are defining the specifications and doing the design. Then the manufacturing aspect is converting the engineering spec into something physical, such as a program. Really, you have to think of systems in terms of products that can be engineered and manufactured like any other product. 'Double-entry bookkeeping' was a system that was first introduced by the merchants of Venice in 1200 A.D. They created the logical system for bookkeeping, but it's been physically implemented many different ways over time: manually, bookkeeping machines, tabulating equipment, and now the computer. But fundamentally, the logical processing is no different than it was hundreds of years ago. So, we have to learn to make that distinction between logical and physical. But you should remember that the technicians are more concerned with the physical than they are with the logical, and that's a serious problem. Logical design must precede physical design. The tail should not wag the dog.

**Q:** WHAT DOES THE TERM 'INFORMATION FACTORY' MEAN?

**BRYCE:** It means setting up an environment analogous to the environment you would have in any manufacturing company. You would have an engineering department for determining your specifications and designs for the products you want to produce. Then setting up a manufacturing environment to build the products accordingly. Of course, you also need a production control department to monitor the assembly lines and a materials management department to standardize and control resources. That's what we have done with the "PRIDE" Information Factory. And just like in manufacturing, we have the ability to change tools and techniques when we want to. This is a function analogous to Industrial Engineering in the

engineering/manufacturing environment.

You know, as simple as these engineering/manufacturing concepts are, organizations such as government, banks, insurance firms and other financial institutions have a real problem assimilating them. It is simply not in their nature to think in terms of products, parts, and so on. Engineering/ manufacturing firms understand these things. But if you've never been trained or educated in this regard, you will be lost in the Information Factory.

**Q:** HOW HAS THE INDUSTRY CHANGED?

**BRYCE:** Well, the industry is still obsessed with the physical aspects of systems. And physically, we've gone from mainframes to minis to PC's. But so what? Even though the physical has changed, the logical hasn't, yet we still ignore it. Again, this is an area where the Japanese excel. They concentrate more on the logical side than the Westerners do and it's starting to show some substantial results, particularly in the types of applications they're producing.

The industry still doesn't know the difference between systems engineering and software engineering. We still call everything software engineering and it's not. Software engineering is writing computer programs. But a system is a logical structure that sits over the physical implementation. A system can be implemented physically many different ways; a computer is but one. By the way, has anyone stopped to think what the word 'application' means? It is the 'application' of the computer to a problem; not the solving of a problem from a systems sense.

I heard that the Department of Defense a few years ago abolished the position of business systems analyst in favor of a software engineering position. This is sad. Instead of having global thinking people, they were content with just techies. I also understand that they're now paying a price for this decision.

**Q:** HOW WOULD YOU DESCRIBE THE DIFFERENCES BETWEEN A SYSTEMS ENGINEER AND A SOFTWARE ENGINEER?

**BRYCE:** A systems engineer is someone who can analyze business problems and design a systematic approach to produce information to satisfy the need. You'll notice that the words 'computer' and 'software' were not mentioned. In the old days, a systems engineer was called a 'systems and procedures analyst'. A software engineer is someone who creates a computer solution to a specified process. In manual systems, there is no such thing as a software engineer, since no computer is involved. The computer brings mechanical leverage to the system processes. Software engineers are fundamentally programmers, and subordinate to systems engineers. Using the term 'engineer' is an attempt at making people think more in terms of a science as opposed to an art.

**Q:** WHAT DO YOU THINK OF THE 'REVOLVING DOOR' POLICY COMPANIES

## HAVE TOWARDS CHIEF INFORMATION OFFICERS (CIO)?

**BRYCE:** The CIO function as advertised in the press is typically staffed using people who have more facade than substance. Have you noticed how many CIO's mentioned in the press have been axed recently? I counted at least five in the last 60 days. The CIO is an excellent concept but should not be staffed by people who spend their time in self-promotion. Maybe this is what irritated their companies.

**Q:** "PRIDE" IS NOW CELEBRATING ITS 20TH ANNIVERSARY. QUITE AN ACCOMPLISHMENT. WHAT DOES THAT MEAN TO YOU PERSONALLY?

**BRYCE:** Well, it means to me that we have found enough people who believe in these advanced concepts to keep us in business for the last twenty years. And I think we're going to stay in business for twenty more years. People ask me why do we spend most of our time overseas and not in the U.S.? Well we do spend a lot of time in the U.S., but we've found the people over here are too software oriented and difficult to train in basic information systems theory. We didn't experience this problem overseas. So, you go where you can best market your product, and this turns out to be places like Japan, Korea, and other places outside the United States.

The English speaking world tends to be somewhat like the United States; whereas the Orient, and particularly Japan, are more oriented to this engineering/manufacturing approach. I jokingly tell people that I will sell outside the U.S. for the next ten to twenty years, and finally it will become obvious to the United States and then I'll spend my waning years selling it here. This is similar to what Deming did with quality assurance.

**Q:** IT ALSO MEANS THAT "PRIDE" HAS STOOD THE TEST OF TIME.

**BRYCE:** Absolutely. I've had people jokingly tell me, who are not technical in the field, that all we're really selling is common sense. They're absolutely correct. It's common sense and it relates to some simple governing principles. But as I've always said, common sense is not very common in this field. Getting caught up in technical aspects tends to blur your perspective. If it's not technically elegant, you lose interest. For example, take the Data Flow Diagram, which is an old technique that actually came out of the 1930's called 'Work Simplification.' All people did was take an old concept and rename it. We've tried to avoid that in our product. But in our field, many who can't invent any new technology just rename old stuff.

Something else; I contend that the Data Flow Diagram doesn't show the flow of data. All it shows is how groups of data are processed as inputs, outputs and files in a particular instance. This is what the old 'Work Simplification' diagrams showed too. But I guess the term 'Work Simplification' wasn't technically elegant enough for some people and the expression 'Data Flow Diagram' sold like hotcakes. To this day I cannot see how it helps you do anything. It isn't even a good way to explain processes to users.

**Q:** WHAT HAVE BEEN THE MOST GRATIFYING ASPECTS TO MBA AND "PRIDE"

**BRYCE:** The most gratifying aspect has been our success. Okay, not 100% success, but we sure have converted a lot of people. Over the years people have told us that "PRIDE" has carried over into their personal lives as well. In Japan, "PRIDE" has become synonymous with climbing the corporate ladder. There have been many Japanese MIS managers who have been bumped upstairs due to "PRIDE". And it's nice to see this when it happens. Another reason why "PRIDE" is successful over there is because it is culturally compatible with their way of thinking and religion. But the fact that we've been active, while others have come and gone, means that we're doing something right.

**Q:** AND WHAT HAS BEEN THE MOST FRUSTRATING ASPECTS?

**BRYCE:** The most frustrating aspect is that MIS managers will purchase the product and not have the self-discipline to learn it and make it work. I don't know why managers think that just by definition of title, they automatically have this knowledge; they don't. Just because you are an MIS Director doesn't mean that you know everything. That's a false sense of arrogance. You should know inside and out the approach you have chosen to manage your organization. If you don't, you're going to go down the drain. It always amazes me when MIS Directors delegate the responsibility for purchasing a management methodology to the techies. Talk about irresponsible behavior.

The other frustrating thing I see is when an MIS Director moves or gets promoted, his or her successor comes in and feels that anything associated with the predecessor is no longer valid and has to be changed. This does not follow. For example, I can recall an insurance company in Canada who used "PRIDE" to document their systems, which was no small job, considering they started out with absolutely nothing. But over a few years the company was able to completely inventory all of their information resources, including data and processing components. Systems were functioning well, projects were coming in on time, and morale was high among the staff. This was the crowning achievement of the MIS Director who eventually retired. Unfortunately, his replacement was a young hot shot who didn't understand the virtue of controlling information resources and saw the company's future rested on the capabilities of Fourth Generation Languages. Consequently, he ordered a purge of "PRIDE" and in a matter of minutes had deleted the company's documented resources off of the computer. Imagine, having all of your company's documented data elements, records, files, inputs, outputs, programs, modules, procedures, systems, etc. all being flushed down the drain. This type of behavior is common in the U.S. and is based on the erroneous belief that anything your predecessor did must be wrong. By the way, you see this type of behavior more in the Western world, than you do in Japan. In Japan, the incoming manager usually adapts to the existing culture, not the other way around.

**Q:** WHY HAS "PRIDE" DONE SO WELL IN JAPAN?

**BRYCE:** This is not surprising. Deming had the same experience with Quality Assurance. He had moderate success in the United States, went to Japan, enjoyed great success, and now you see the results: quality products, quality service, etc. Deming is a hero in Japan, and now only twenty to thirty years later are U.S. companies starting to understand what he was talking about. Well, the same is happening in the field of IRM with us. "PRIDE" has had the same kind of reaction in Japan. Whereas Japanese companies see systems development as a science, their American counterparts see it as an art. And don't kid yourself, the Japanese have every intention of dominating the systems and software market in the same manner as they have taken over in other markets, such as automobiles, steel, electronics, banking and so on.

**Q:** WHAT PROGRESS HAS BEEN MADE OVER THE LAST 20 YEARS IN SYSTEMS DEVELOPMENT? ARE WE MOVING FROM AN ART TO A SCIENCE?

**BRYCE:** Well, you are moving towards a science if you've been using "PRIDE". This alone means that you believe that there are some governing principles for building systems. Unfortunately, we do not corner the market with our approach. There are other approaches that are based on art. So, I believe overall that the industry in the U.S. sees systems development as an art, not as a science. This is one of the major problems we want to overcome with the "PRIDE" Information Factory. We're going to attack with a religious fervor. We're going to try to demonstrate to the field that a science does, in fact, exist and that it can be taught using common engineering/manufacturing principles.

Recently, I was reading about the so-called 'Software Factories' in Japan. Obviously, they believe as we do that you can design and build quality software like any other product. They're going to master this and then raise havoc in the Western world.

**Q:** HOW IS THE INDUSTRY IN TERMS OF STANDARDS?

**BRYCE:** Our industry is a disaster. Back in 1970 I gave a talk on standards to DPMA in Seattle. I created a bit of a controversy by challenging the association to get more involved with instituting industry standards. But you know something? The industry has not made one inch of progress since then. The industry has no standards. Even academics in the field do not define their terms well. For instance, one of the things that really bothers me is when people talk about information like they know really what it is. They don't! Try looking in 'Information Engineering' related books and brochures. You're not going to find it. The terminology and thinking in our industry is just plain sloppy.

People still believe that systems and software are synonymous. They most definitely are not. Some think IBM's AD/Cycle is a way of developing systems. It's not. It's only a way of developing programs using their tools. It's a programming environment. It doesn't help you with systems engineering. IBM knows this. IBM sells computers and programming tools. They don't sell systems design and development tools. AD/Cycle is nothing more than the old five step 'waterfall approach' for managing



software projects.

**Q:** IN THIS AREA, IS GOVERNMENT A CATALYST OR IMPEDIMENT TO PROGRESS?

**BRYCE:** Well, I used to be on some government sponsored programs, such as Data Base Directions, and on some of the standards committees. The results of these were so minuscule and primitive that I abandoned trying to work on these committees. I don't think government enhances anything from that perspective. The government may try, but not with any vigor. The government should develop more standards, particularly since they're the world's greatest user, but you don't find any consistency within the government at all.

**Q:** HOW DO YOU FEEL ABOUT INDUSTRY CERTIFICATION PROGRAMS? ARE THEY WORTH THE PAPER THEY ARE WRITTEN ON?

**BRYCE:** I think we can come up with some certification for people such as programmers or someone like that since we're talking about techniques. But I have trouble with certification in the systems area. Most of these certification programs are from a programming point of view and not the business systems engineering view. We've hired people who have been CDP's and had Computer Science degrees and we've had mixed results. A lot more development is required in this area.

**Q:** HOW ABOUT CERTIFYING UNIVERSITY CURRICULUM?

**BRYCE:** I have trouble with universities. You have courses designed by people who attended school, graduated, then went on to graduate school and became teachers without any real-world experience whatsoever. I feel that when academics talk to other academics and quote each other, it's like multiplying zero times zero, you still end up with nothing. I'm not terribly impressed with what comes out of universities. I think we have a lot of academic hullabaloo. For example, I was in this business way before the academics even knew there was such a business. We were developing things in industry way before the academics. Remember, necessity is the mother of invention. I think the universities have a long way to go in the area of Information Resource Management. The universities should staff with practicing professionals like they do in Law and Medicine.

You know, when you hire someone with a college degree in this area you have to first debrief them and re-educate them in the ways of the real systems world. This would be a lot easier on business if the universities had their act together.

**Q:** BACK IN 1979 MBA WON A LANDMARK COURT RULING INVOLVING THE MISAPPROPRIATION OF TRADE SECRETS. HOW WOULD YOU DESCRIBE THE ETHICS IN THE INDUSTRY?

**BRYCE:** I don't think our industry has anywhere near the ethics that it should have.

There's no question that many of the consultants have no trouble whatsoever of taking things that they learned at a customer's site or learned using other people's products and adopting them for their own use. They don't have a great deal of respect for the proprietary interests of others. They feel that's an interference. Our lawsuit clearly showed that. The firm we were suing was so arrogant, they felt we had a lot of nerve trying to stop them from using our proprietary know-how. I see this all the time. They're out to earn money by selling their services by the hour. I think any knowledgeable person in this field should look at what some of these so-called 'System Integrators' have done. Whether they have successfully delivered the products they are talking about or whether it was just an excuse to rip off the client. I remember the New Jersey Motor Vehicle fiasco some time ago. I think that was just unprofessional and irresponsible behavior. But it happens in this industry just because the company auditors think they are experts in the area of systems development. This, of course, is not true. They should be held accountable to a higher standard. When you see these people doing fixed price contracts consistently, then you'll know that their working on a standard basis. But when its on a time and materials basis, I say 'caveat emptor!'

**Q:** DOES THE INDUSTRY SUFFER FROM 'YELLOW JOURNALISM'?

**BRYCE:** Back some years ago there was a magazine that was called the 'EDP ANALYZER'. It has since been replaced by something else. But I do recall that when this magazine came out they attempted to do real research and validate their stories. There was no advertising in it. They tried to be as technically accurate and honest as possible. Now I see in the magazines nothing more than what I consider hype. I know first hand some companies and applications that were real disasters. Yet, when they come out in print, the press tries to hype their success. For instance, right now on CASE tools, you hear all kinds of success stories. But I think if you went out and investigated you would find some rather shallow results. You will also probably find that nine times out of ten, the articles are written by the vendors. So I don't think our magazines are serving us well. I think many of the stories are nothing more than marketing hype; to do nothing more than to promote advertising in the magazine. I just have real trouble with what's going on. CASE tools are a classic example of this. You also see it in the areas of relational DBMS tools, 4GL's, etc. We have to have more honesty and integrity from these magazines, otherwise we'll never get to the truth.

The press is also indicative of what's going on in the industry. Whereas you used to have credible publications like the 'EDP ANALYZER' and 'INFOSYSTEMS' which tried to serve the systems field, they have all gone by the wayside and have been replaced by computer hardware/software publications. Guys like Dick Canning, Arnold Keller and Wayne Rhodes did a hell of a good job. Yet, the industry turned its back on them.

I have also been reading a great deal in the press about the industry's history. I've personally experienced a good bit of the history firsthand. I've seen some relative newcomers talking about history and their facts are not correct. One of the most ridiculous things I've recently seen was a trivia section where the author asked 'What is a punchcard?' His answer, 'an 80 column card', was totally incorrect. The first

punchcard came out from the Bureau of Census in 1890 and it was 45 columns, not 80 columns. The point is, what he was saying was only partially correct. Yet, this is the typical behavior by many people in the industry; they recall history to suit their needs. This is really disturbing to me. We simply do not have a good sense of history. Consequently, the real pioneers, achievers, and innovators are forgotten. One of the things I find disconcerting is that people who have only been associated with IBM computers also have a really distorted sense of history. Not only do they have a distorted sense of what occurred, but so does IBM. I remember what they said about Virtual Memory, they thought they invented it, when actually it was invented by Burroughs a long time ago with the B-5500; way before IBM. So, I take most of this historical perspective with a grain of salt.

**Q:** WHY DID YOU WRITE THE 'IRM REVOLUTION' BOOK?

**BRYCE:** We wrote the book for several reasons. Number one, we wanted to set the "PRIDE" concepts and philosophies in the public domain. When we were only selling a manual product, we needed to protect our concepts. This is why we had the lawsuit with the CPA firm. But now this is not important. We want everyone to understand the concepts and philosophies in order to promote the Information Factory. The industry, in general, is very backward and we wrote the book to help bring it out of the 'stone age.'

As an aside, the book is a best seller in Japan. I would sure like to see it become a best seller over here, but maybe that will be five to ten years from now. However, we are encouraged to see some universities beginning to use the book in their curriculum, so maybe there's hope.

**Q:** THE COMPANY WAS ORIGINALLY FOUNDED IN CINCINNATI, OHIO. WHY THE MOVE TO THE TAMPA BAY AREA OF FLORIDA IN 1985?

**BRYCE:** As a Northerner I was becoming fed up with the harsh winters. Since we were already running on an international scale, it became rather obvious that we could run our business from anywhere. So, using "PRIDE", we developed our requirements and began researching the country, from California to the East Coast. We looked as far north as North Carolina and as far south as Florida and Texas. We looked at Sacramento, San Diego, New Mexico, Arizona, Austin, Houston, Myrtle Beach and places like that. We looked at southern Florida but then found out about the Tampa Bay area from one of our customers in the area. Here, we found the answer. It met our specifications, including a modern international airport, easy access for our employees, and a place where our customers would want to come for training. Palm Harbor, Florida met all of our needs and we moved. We have never been sorry about that decision. This is an excellent place to conduct business and live.

Don't get me wrong. Cincinnati was a great place to start the business. Its geographical location was excellent for serving the U.S. mid-west, east coast and Canada.

**Q:** WHAT LIES AHEAD FOR "PRIDE"? WHAT IS IT'S FUTURE?

**BRYCE:** We feel our future now lies in the "PRIDE" Information Factory. This is the next evolutionary step in the development of "PRIDE". We are developing a whole approach under IBM's OS/2 and we're implementing the entire Information Factory with LAN networks. We're gambling that this is the wave of the future. Naturally, we think we're absolutely correct. We think that systems will be developed on LAN networks using personal computers. So, we're putting the whole Information Factory on this platform. This will make "PRIDE" suitable for all companies, from the smallest to the largest. We also believe that our field will come to see CASE tools as limited solutions and will have an appreciation for this more comprehensive approach.

Up to now our approach was intellectually correct but lacked the sex appeal. Now we're adding the sex appeal as provided through the personal computer. In other words, we're getting around to putting facade on to the substance. And this can mean nothing more than success.

Back in 1974 when we started to build the supporting "PRIDE" software, we were concerned with producing a generic product that would operate the same on all mainframe platforms. Although this provided customers with portability, it somewhat limited our ability to produce flashy inputs and outputs. Now with the proliferation of the personal computer, with its graphical capabilities, we are now going to add the flash that our customers want.

**Q:** YOU HAVE ALWAYS MADE A CLEAR DISTINCTION BETWEEN "PRIDE" AND CASE; THAT IT IS A 'CASE-COMPLEMENTARY' APPROACH. DO YOU EVER FORESEE IT GOING AGGRESSIVELY INTO THE CASE MARKET?

**BRYCE:** We always think of the "PRIDE" Information Factory as much larger than just software engineering. We do have aspects in our product where we do software engineering and we're moving into that area more rapidly. We're putting some of our own CASE tools into the Information Factory. Bear in mind, techniques and tools are things that come and go, so we've still engineered our product as a framework where if better techniques and tools come on the scene we can easily incorporate them into the product. We will provide our CASE tools but will allow other CASE tools to be used in the Factory. This is what we refer to as having an 'Open Architecture'. But our forte will still remain to provide the environment, the culture, and the management tools for developing integrated corporate-wide information systems.

**END**

**COPYRIGHT (C) MBA 1995**

# ***BRYCE'S LAWS ON INFORMATION RESOURCE MANAGEMENT***

Productivity = Effectiveness X Efficiency

Information = Data + Processing

There is nothing more unproductive than to build something efficiently that should not have been built at all.

Organizations progress when the impact of good actions and decisions outweighs the impact of poor actions and decisions.

IRM is the view of the enterprise from 50,000 feet.

We must apply the same discipline, organization and automation that we recommend for other parts of the company.

IRM requires the creation of a new environment with an engineering/manufacturing orientation. This will require retooling and retraining.

Technology alone will not solve our problems, only effective management will.

No amount of elegant programming or technology will solve a problem if it is improperly specified or understood to begin with.

If anything in life is constant, it is change. The only constant involved with information is that it is seldom static.

If an information requirement is stated improperly to begin with, then everything else that follows will be incorrect.

The only way that information systems communicate, both internally and externally to other systems, is through shared data.

Data is stored, Information is produced.

Quality must be built into the product during design, not inspected in afterwards.

Enterprises with identical missions will also be identical in terms of logical structure.

Whereas logical IRM resources will remain relatively static, the physical resources will change dynamically.

Information is a perishable commodity; it only has value at a particular point in time.

Information is highly volatile in that it is greatly influenced by external factors, such as government, economics, competition, customers, etc.

An elegant solution to the wrong problem solves nothing.

The day a company goes into business is the day when its information systems are born.

Only when the systems analyst can walk in the moccasins of the user does the analyst have a right to design a system for the user.

Information is for people, not for the computer.

An information system is a product that can be engineered and manufactured like any other product.

Systems  $\neq$  Computers  
Systems  $\neq$  Software  
Systems  $\neq$  Projects

All information systems have the same structure. In manufacturing terms, it is known as a "four-level bill of material."

Systems are designed by 'explosion' and implemented by 'implosion.'

No one has ever built the perfect system the first time, and no one ever will.

Systems are built by evolution; not by revolution. The day when a system is installed, is the day it begins to undergo change.

Good Systems Design + Good Programming = Great Systems  
Good Systems Design + Bad Programming = Good Systems  
Bad Systems Design + Good Programming = Bad Systems  
Bad Systems Design + Bad Programming = Chaos

How a system is implemented is of little importance if it solves the problem effectively.

Programming is a translation function, going from human understandable specifications to machine readable instructions.

Good specifications will always improve programmer productivity far better than any programming tool or technique.

Beware of your "firefighters," they are probably your chief arsonists.

The first on-line, real-time, interactive, data base system was double-entry bookkeeping which was developed by the merchants of Venice in 1200 A.D.

All organizations have a data base; some are managed, most are not.

You must first plant the seeds in order to harvest the crop. Unfortunately, most companies tend to eat the seed and then there is no crop to harvest.

There is only one problem with common sense; it's not very common.

A methodology is nothing more than an assembly line that produces a finished product.

Project management is only possible with an effective methodology.

A methodology that simply keeps employees busy and doesn't produce anything is an exercise in futility. This type of methodology is commonly referred to as "The Dance of the Fairies."

Systems do not have a "life cycle." They may go on forever if kept viable with change. The only thing that has a "life cycle" is a project which has a beginning for planning, a middle for execution, and an end for review.

It is one thing to enact legislation, it is another to enforce it.

Project management is a philosophy of management, not a tool or technique.

Most estimating errors are errors of omission, not commission. It is what we forget to estimate that gets us into trouble.

An estimate improves in accuracy in relation to the level of detail considered.

A project always has a critical path until it is completed or closed. The path can vary depending on accomplishments.

A project requires a methodology, but a methodology does not require a project.

It is just as bad to underrun a project as it is to overrun a project.

If we lived in a perfect world, there would not be a need for managers; projects would be executed on time and within cost. However, the reality is, we live in an imperfect world.

Managers do not create problems, they solve problems.

Manage from the bottom up; not just from the top down; this creates personal commitment and accountability.

Employees should be treated as professionals and held accountable for their actions.

All companies have a culture. In order for employees to function and succeed, it is essential they understand and believe in the culture.

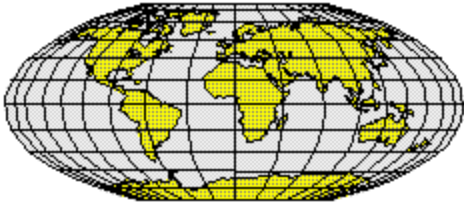
Time lost, is time lost forever. You cannot buy it back.

END

COPYRIGHT (C) MBA 1995



## ***ABOUT MBA***



### **M. BRYCE & ASSOCIATES, INC.**

777 Alderman Road  
Palm Harbor, FL 34683  
United States of America  
Tel: 813/786-4567  
Fax: 813/786-4765  
BBS: 813/786-4864  
E-Mail: [TimB1557@aol.com](mailto:TimB1557@aol.com)  
IBM Link: DEV2643

Since 1971: *Software for the finest computer - the mind.*

In Japan:

### **PRIDE Japan, Inc.**

Masujima Building  
8-13, 1-Chome  
Higashi Gotanda  
Shinagawa-Ku, Tokyo 141  
Tel: 3/3280-0411  
Fax: 3/3280-0418

### **CORPORATE PROFILE**

M. Bryce & Associates, Inc. (MBA) is an international management consulting firm specializing in Information Resource Management (IRM). To this end, the company develops and markets products and services aimed at IRM. The company's "PRIDE" product line includes methodologies, techniques and tools, which have been used worldwide in every field of endeavor imaginable:

#### Countries Served

Australia  
Brazil  
Canada  
Denmark  
Japan

#### Industries Served

Aerospace  
Automotive  
Banking  
Chemical  
Communications

Korea	Construction
Mexico	Electronics
New Zealand	Energy
Norway	Engineering
Saudi Arabia	Government (Federal/State/Local)
South Africa	Insurance
Spain	Manufacturing
United Kingdom	Paper/Wood
United States	Printing/Publishing
Venezuela	Public Utilities
	Retail/Wholesale
	Shipbuilding
	Steel
	Transportation

In particular, "PRIDE" products have been used extensively by firms throughout Japan, including several companies who have received the prestigious Deming Prize for quality.

Since 1971, MBA has never failed to meet a customer implementation commitment, regardless of the customer's geographical location or training requirements.

MBA is totally focused on the science of Information Resource Management. IRM is our only business. We have invested over 23 years of research and development into our product line, constantly upgrading and fine-tuning the product line to suit customer and industry requirements.

*"(MBA's) strategy has included consistent, long-term investment in research and development."*

- DATAPRO

Through the "PRIDE" product line, MBA has established several precedents and introduced many new concepts to the industry:

1. **First** to offer commercial methodologies for system design, data base design, and enterprise engineering.
2. **First** commercial repository for capturing, controlling, and re-using information resources.
3. **First** on-line methodology for accessing automated instructional materials.
4. **Concepts and Techniques introduced:** Information Driven Design, Structured Systems, Chronological Decomposition,

Layered Documentation, Data Resource Management, 4 Data Base Models, Enterprise Decomposition, Priority Modeling, Bill of Material Processing for managing information resources, Mini-Project Manager, Estimate-to-do (versus percent complete), estimating by bill-of-materials.

While many of MBA's competitors have come and gone, the "PRIDE" product line enters its third decade of use. This is a testament to the integrity and durability of the product and the company who produced it.

*"With almost 1,500 installations worldwide and the fact MBA has existed in the methodology world for almost 20 years, it is fair to say that the company is well capable of managing its revenues and balancing them off with consistent research and development by investing about 80 percent of those revenues into R&D."*

- DATAPRO

END

COPYRIGHT (C) MBA 1995

## *MBA Product Offerings*



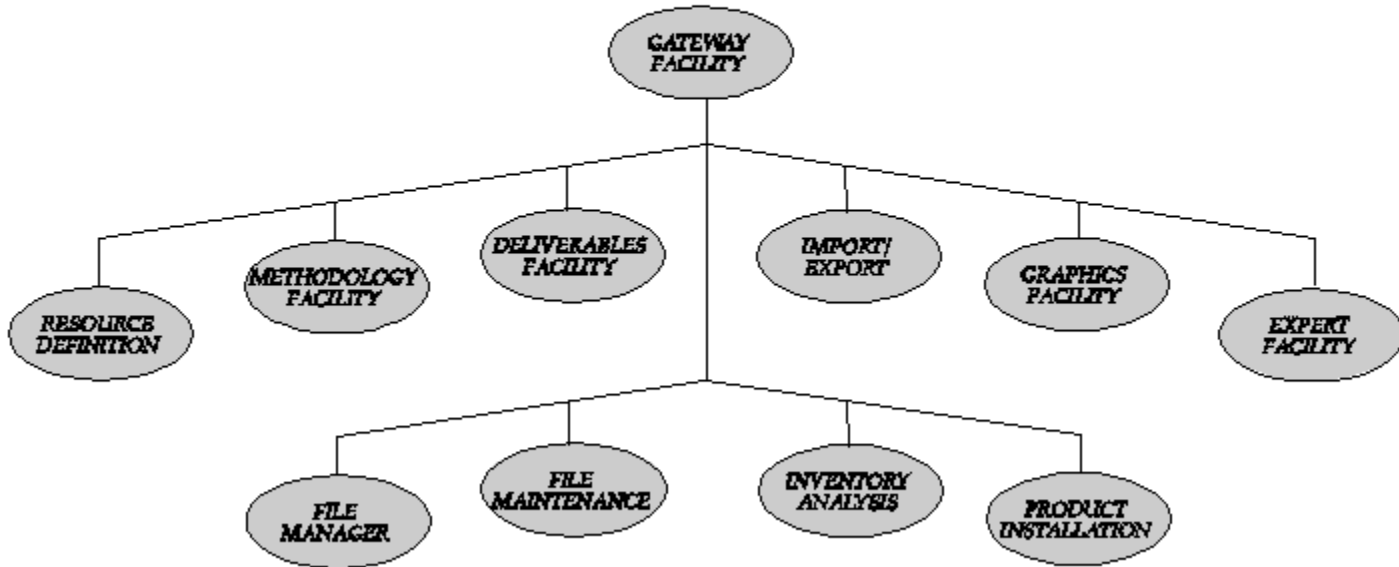
The flagship of MBA's product line is the **"PRIDE" INFORMATION FACTORY** an OS/2 based product for managing information resources. It includes:

1. Planning and design tools as used in support of the "PRIDE" methodologies (EEM, ISEM, and DBEM).
2. The IRM Repository where information resources are cataloged and inventoried. The IRM is the engine of the FACTORY software; all of its tools interface with the IRM to collect and analyze resources. Further, the IRM has an "open architecture" enabling it to interface with a wide variety of third party software packages (e.g., CASE tools, DBMS packages, spreadsheets, word processors, program generators, other repositories/data dictionaries, etc.).
3. An integrated Project Management system that works with the "PRIDE" methodologies as well other user defined Work Breakdown Structures (WBS). The system offers tools in support of project planning, estimating, scheduling, reporting, and control.
4. The same "PRIDE" methodologies are embedded in the FACTORY software, allowing on-line access to narratives and examples, as well as printed documentation. In addition, the customer can add supplemental in-house standards thereby providing the means to maintain all standards documentation electronically.

### **PRODUCT ARCHITECTURE**

The "PRIDE" INFORMATION FACTORY itself was engineered and manufactured using earlier versions of the "PRIDE" Family of Products. In essence, "PRIDE" begat the next generation of "PRIDE." As a "PRIDE"-built product, it consists of sub-systems, inputs, outputs, files, etc. like any other system developed under "PRIDE." There are a total of eleven sub-systems in the product.

# PRIDE® INFORMATION FACTORY™ ARCHITECTURE



## GATEWAY FACILITY

This represents the formal entry point to the product. Users logon through the Gateway where their product privileges are checked. After entering, they can modify file and printer assignments, call for their time screen to record their hours worked against project assignments (assuming the Project Management system has been enabled), or access the other FACTORY facilities (depending on their privileges).

## RESOURCE DEFINITION FACILITY

This facility is used to search, display, edit, and print information resources. A user may search the IRM using six different search techniques:

1. Search by Resource Name.
2. Search by Resource Control Number.
3. Search by Logical Attributes (synonym words).
4. Search down a Chain of resource relationships (e.g., display data elements subordinate to a file or data base).
5. Search through a Data Taxonomy for data descriptions.

6. Search for data resources based on Program Label.

RESOURCE NAME SEARCH

SELECT RESOURCE TYPE:

ENTER ALPHANUMERIC STRING (\* IS WILD):

DATA DESCRIPTIONS (DD)  
RECORD DESCRIPTIONS (RD)  
INPUT DESCRIPTIONS (ID)  
OUTPUT DESCRIPTIONS (OD)  
FILE DESCRIPTIONS (FD)

PROD\*

Confirm Cancel Display Help

RESOURCES FOUND.: 8

NUMBER	NAME
DD-00003	PRODUCT NUMBER
DD-00123	PRODUCT SALES PERFORM - MONTH
DD-00121	PRODUCT SALES PERFORM - YTD
DD-00176	PRODUCT STANDARD COST
DD-00175	PRODUCT UNIT SHIP WEIGHT
DD-00162	PRODUCT UNITS REQUISITIONED
DD-00119	PRODUCTS SALES PERFORM - YTD
DD-00116	PRODUCTS SALES PERFORM-MONTH

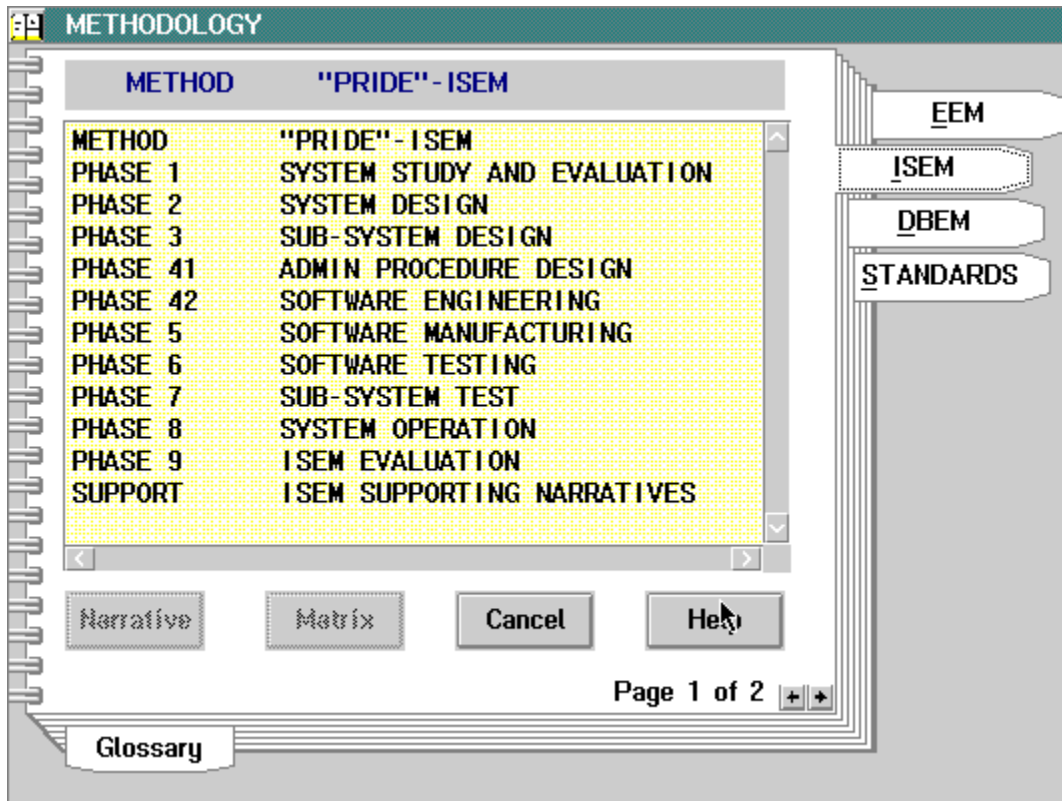
As an engineer enters specifications about the various information resources, field entries are validated by the software. HELP facilities are, of course, available.

## METHODOLOGY FACILITY

This facility provides guidance when executing the various "PRIDE" methodologies (EEM, ISEM, DBEM). There is an interactive portion which provides instructions on how to perform the phases and activities of the methodologies. These instructions are also accessible through other sub-systems, such as the Resource Definition Facility, Graphics Facility, Deliverable Facility, Expert Facility, and the Import/Export Facility. A glossary of "PRIDE" related terms is also available interactively.

For those customers wishing to produce standards manuals, this facility provides the ability to generate "PRIDE" manuals, complete with text, examples, and worksheets. The manuals can be produced in their entirety or in portions (such as a phase at a time).

The "PRIDE" methodologies are maintained in the AIM Master File (Automated Instructional Materials), a computer accessible file. In addition to "PRIDE" specific text, the customer can add supplemental standards to the file. For example, the customer may wish to add text regarding the use of special in-house tools, techniques, naming/numbering conventions, etc. In this way, AIM is used to manage all corporate development standards.



## DELIVERABLES FACILITY

All of the various reports produced as a result of following the "PRIDE" methodologies can be produced automatically. The user has the option to build whole reports or request selected outputs. The types of methodology deliverables include: Feasibility Studies, System Design Manuals, Flowcharts, User Manuals, Computer Run Books, Program Specifications, Data Base Designs, Audit Review Reports, Checklists, etc.

Various Project Management reports are also included: Time Distribution Summaries, Project Progress Reports, Project/Personnel Estimates/Schedules, Effectiveness Rate Charts, Gantt Charts, Resource Allocation Charts, Skills Inventory, User Billing/Chargeback, etc.

## IMPORT/EXPORT FACILITY

This represents the formal interface between the FACTORY and third party software products. The "importer" consists of a converter routine to take an externally prepared

data file and either reformats it or validates it in preparation for updating the IRM Repository. The "exporter" allows the customer to extract information resources from the IRM. The data can then be manipulated as required. For example, a user can export resources from the IRM to a relational Data Base Management System (DBMS) which can then be used to massage the data and build special reports. In this way, the user is provided with the capabilities of a report writer. Data can be exported in a variety of formats:

1. Text files for use in text/word processing packages.
2. Delimited ASCII format (.DEL) for use in relational data base management systems and other programming aids.
3. Standard 80 character records which can be edited and reapplied to the IRM Repository.
4. A user defined format to accommodate any type of output.

The INFORMATION FACTORY also makes available callable program modules which allows customers to develop seamless interfaces between the FACTORY and other external programs, thereby providing the ability to directly interrogate the IRM and retrieve data from it. Customers can use this facility to build bridges to a variety of software tools, such as DBMS packages, CASE tools, program generators, report writers, etc.

## GRAPHICS FACILITY

This sub-system provides the ability to draw "PRIDE" related graphics at the terminal and populate the IRM Repository at the same time. For example:

**Hierarchy Charts/Indented Lists** - including business function charts and organization charts.

**Matrices** - to represent relationships between different types of information resources.

**Flowcharts** - for Systems, Sub-Systems, and Computer Procedures.

### Information Flow Diagrams (IFD)

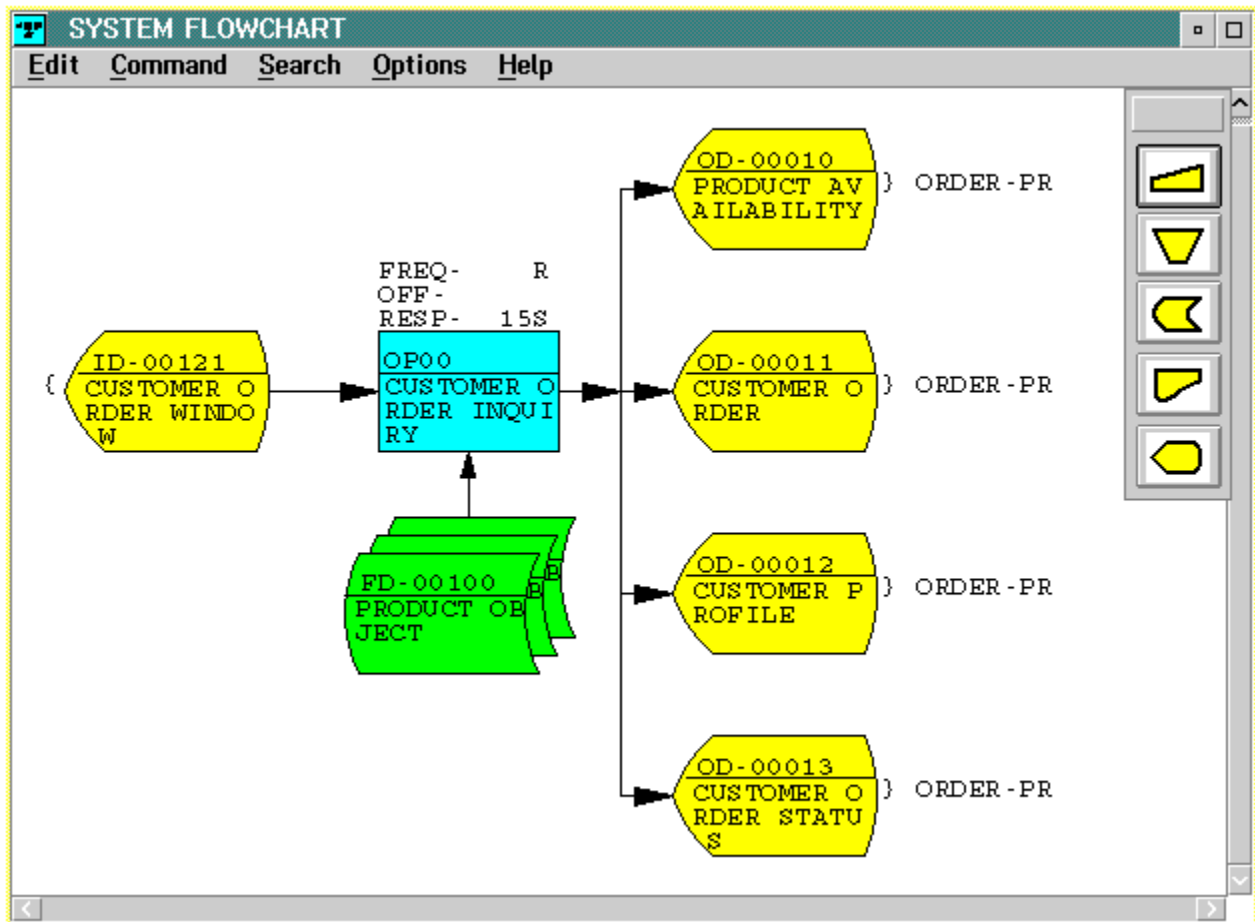
These graphics are not stored as bitmaps or some other graphical format. Instead, they are dynamically drawn in accordance with the specifications maintained in the IRM Repository. As a user establishes relationships between two or more resources in one diagram, it will be reflected in other diagrams automatically. For example, should one user change the relationship of two resources in a hierarchy chart, the change will be seen by another user working with an indented list. MBA refers to this approach as "interpretive graphics." It means that graphics are based on specifications maintained



in the IRM and, as such, documentation is kept current and up-to-date; as a resource relationship changes, so does the documentation.

The user may print the graphics as desired (to a PostScript supported printer) or copy them to the OS/2 Clipboard for use in other applications or packages.

### SAMPLE SYSTEM FLOWCHART



### EXPERT FACILITY

This facility provides special routines to expedite the administrative tasks associated with the "PRIDE" methodologies. Features include:

An **"Impact Analysis"** feature to study the effect of a proposed change to a resource, such as a data element, business function, or system. Acting as a bill of materials processor, the "Impact Analysis" can generate a list of resources effected by the change. This feature alone has proven invaluable to companies. It permits them to play "what if" and allows them to make intelligent project decisions by providing a complete "roadmap" of all required resource changes.

For "PRIDE"-EEM, an **organization analysis** aid is provided, along with a priority modeling tool used to automatically calculate the priorities of corporate objectives and supporting projects.

"PRIDE"-ISEM, a **Computer Aided Design (CAD)** tool is provided to automatically design enterprise-wide information systems (perfect for Business Process Re-Engineering).

"PRIDE"-DBEM, various data base design aids are provided.

"PRIDE"-PM, special aids are provided for planning, estimating, scheduling, and reporting.

### IRM FILE MANAGER FACILITY

This facility provides the means to audit, expand, reorganize, and correct the primary files associated with the product. This primarily consists of the IRM Master File and its subordinate files. It is also used to set the basic operating parameters of the file (e.g., file size, print options, PM switch, DBCS switch, etc.).

### FILE MAINTENANCE FACILITY

This sub-system represents the principal means for applying changes, additions and deletions to resource definitions in the IRM Master File. All file maintenance activities includes a rigorous data validation check to assure resources are being defined according to standards.

A "Resource Control" feature is provided for the "PRIDE" Coordinator to "activate" (freeze) resource definitions, thereby prohibiting further changes to the definitions. As systems are completed and placed into production, the "PRIDE" Coordinator turns all of the associated resources into an "active" status. As it becomes necessary, the coordinator can also use the facility to place pertinent resources back into a "development" mode, thus allowing modification to the definition.

### INVENTORY ANALYSIS FACILITY

Control logs are provided to serve as a manifest of an enterprise's information resources. Resources are listed by type, and in either control number sequence or by name. These logs also show redundant resource names, whether the resources are "not used" (not attached to other resources, and whether they are in an "active" status (production mode) or under "development." A summary report is provided reviewing the development status of all resources in the organization. In addition, special reports are provided to analyze such things as program labels, forms, computer schedules, etc.

### PRODUCT INSTALLATION

This sub-system is used to initially install the INFORMATION FACTORY, to setup additional users and workstations, and to implement product upgrades. The product is well packaged and easy to install. Detailed instructions are provided through the computer which guides the customer through the installation process.

## **TECHNICAL REQUIREMENTS**

The "PRIDE" INFORMATION FACTORY was designed to be compatible with IBM's Systems Application Architecture (SAA). As such it was designed according to rigorous standards to give it the same "look and feel" as other SAA/CUA (Common User Access) compliant products. This makes it easier to learn and implement.

The FACTORY can be implemented either as a standalone product, or as a client/server application.

OS/2 Version 2.x (or higher) is required on an Intel 486 based computer (or higher). 8mb of memory (or higher) is required for workstations, 16mb of memory required for a file server (or standalone). All OS/2 supported Local Area Networks (LAN) can be used. Text reports produced from the "PRIDE" software support all OS/2 supported printers. Graphical reports require "PostScript" supported printers. Consult the vendor for complete technical specifications.

The product makes use of its own proprietary file management system and is, therefore, independent of third party DBMS packages. Programs are written in "C" and "COBOL" and make use of the OS/2 "Presentation Manager" Graphical User Interface (GUI).



## **TRANSLATION CONSIDERATIONS**

All of the screens, reports, and messages associated with the "PRIDE" INFORMATION FACTORY can be translated to suit a particular language. Printed maps and messages can be edited as required. The vendor can also provide the screen panels for translation. The product provides for the Double Byte Character Set (DBCS), making it suitable for use with Japanese and Chinese character sets (Kanji).

## **MATERIALS, SERVICE AND PRICING**

Included in the purchase price is an implementation guide and executable programs.

Training and consulting services are provided for the development staff, users, and executives. The product is normally sold on a site license basis. Consult the vendor for full pricing details.

**FOR ADDITIONAL INFORMATION:**

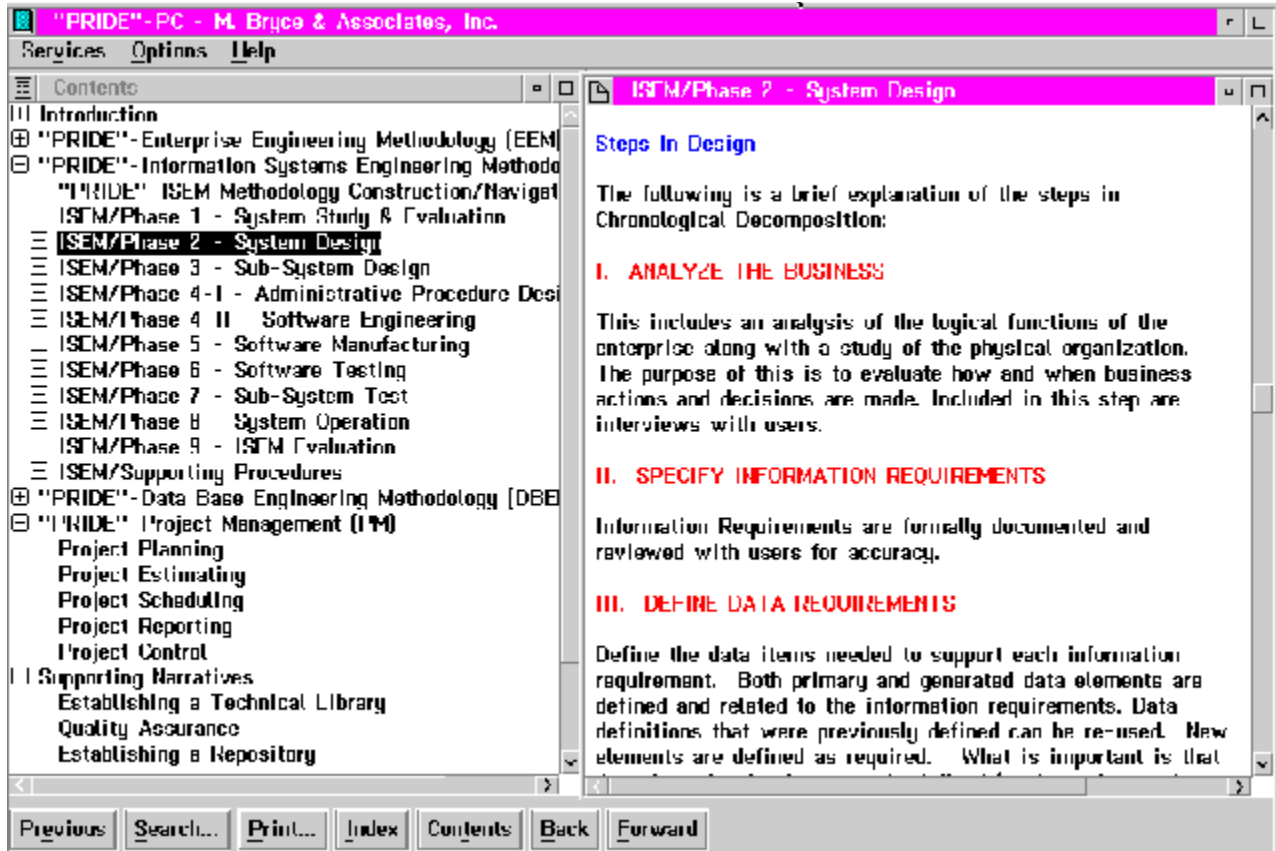
To learn more about the "PRIDE" INFORMATION FACTORY, either contact the vendor...

About MBA

...or reference the following sources:

IBM National Solutions Center  
OS/2 CD-ROM of ISV Applications  
The OS/2 Development Tools  
OS/2 Exploiting Applications Directory  
DATAPRO  
Data Sources  
And other fine software guides.

***"PRIDE"-PC***



### ***Thorough/Complete/Integrated/Proven***

For those who need proven methodologies, at an affordable price, comes "PRIDE"-PC , a desktop reference for the "PRIDE" methodologies, including:

Enterprise Engineering  
 Information Systems Engineering  
 (integrating BPR with Software Engineering)  
 Data Base Engineering  
 Project Management

With "PRIDE"-PC, you can display, search, index, and print the "PRIDE" methodologies from your PC. The product includes:

- \* Tutorials and instructions with hypertext
- \* Samples of deliverables
- \* Review Checklists (specifying acceptance criteria of deliverables)
- \* Functional Matrices (shows Who is to perform What, When)
- \* Supporting narratives (including sample job descriptions)
- \* Glossary of Terms

Now you can have a desktop reference to your methodologies instead of voluminous

"shelfware" manuals gathering dust. And at a price of just \$150 per PC, why pay more? "PRIDE"-PC requires **OS/2 2.x** (or higher) or **MS Windows 3.x** (or higher).

***"No developer or IS manager should be without "PRIDE"-PC."***

**NOTE:** Prices are in U.S. dollars and subject to change without written notice.

**END**

**COPYRIGHT (C) MBA 1995**

## *MBA Service Offerings*



In addition to its "PRIDE" product line, MBA offers a full line of consulting and training services. This specifically includes the following areas:

### **Methodology Customizing Training and Education General Consulting**

#### **METHODOLOGY CUSTOMIZING**

The methodologies contained within "PRIDE" are universally applicable and independent of any particular tool or technique. However, there may be occasion where the customer may wish to customize "PRIDE" to accommodate specific requirements. In such an event, MBA can be contracted to develop a tailored version of "PRIDE". Potential areas for development and inclusion in "PRIDE" include:

1. **Policies and Procedures.**
2. **Job Descriptions.**
3. Specific **Tools & Techniques** to be used throughout the "PRIDE" methodologies and Project Management system. For example:

**Performing a Cost/Benefit Analysis** (including standard formulas for calculating Return on Investment and Break Even Points).

**Program Design** - such as structured programming, object oriented programming, diagramming techniques to be observed, etc.

**File Design** - to develop certain types of physical file/DBMS structures (e.g., hierarchical, network, relational, object oriented, VSAM, etc.).

**Writing Standards** - for developing narratives, help text, user documentation, etc.

**Testing Standards** - for uniform testing criteria.

**Naming and Numbering Conventions** to be observed.

**Program Coding Standards**

**Auditing Standards**

**Change Control Standards**

**Conducting a Meeting**

Use of a specific **CASE tool**

**Documentation Aids** - e.g., diagramming aids, source code librarian aids, etc.

**Compilers**

**Testing/Debugging Aids** - including test data generators.

**Editors**

**Application Development Aids** - such as program generators, 4th generation languages, and report writers.

**Prototyping Aids** - such as screen painters, and Graphical User Interfaces (GUI).

**Repositories/Encyclopedias/Data Dictionaries**

**Project Management Aids** - such as use of Gantt Charts, PERT diagrams, and other tools for planning, estimating, scheduling, reporting, and control.

4. **Corporate Tailoring** - to include a corporate name, logo, and in-house standard deliverables (examples).

5. **Translation of the Product**

Methodology customizing is performed by MBA on a time and materials basis. Please discuss your requirements with MBA and let us quote you a price to tailor "PRIDE".

**TRAINING AND EDUCATION**



MBA offers a course curriculum oriented towards IRM related disciplines. There are three types of training courses available from MBA:

1. **Skills Training** - aimed at teaching the basic skills required to perform Information Resource Management, Enterprise Engineering, Information Systems Engineering, Data Base Engineering, and Project Management. This series of courses does not make use of any computer software. Instead, it combines instructor lecture with active student participation. Most courses involve practical workshop exercises involving teamwork. Such team exercises promote group dynamics and the exchange of ideas between students, regardless of their level in the corporate structure.
2. **"PRIDE" Software Training** - aimed at teaching the effective use of "PRIDE" products, such as the "PRIDE" INFORMATION FACTORY. Lab exercises include use of the computer software.
3. **"PRIDE" Methodology Implementation** - combined education/consulting aimed at facilitating the effective implementation of the "PRIDE" methodologies (EEM, ISEM, or DBEM). This is particularly useful when initiating the use of "PRIDE."

All courses are conducted in either a classroom or "roundtable" environment. Each student is provided with a workbook that includes such things as printed copies of the transparencies, workshop exercises, a glossary of terms, and a place for notes.

Most MBA courses include a questionnaire at the end of the course to test the student's knowledge. In addition, an evaluation of the course is performed by both the students and the instructor. Consequently, the customer is provided with an analysis of the course at no extra charge.

Having trained thousands of managers, analysts, engineers, and programmers from around the globe, MBA has extensive educational experience. Our instructors are seasoned professionals with "hands-on" experience in the use of IRM concepts, methods, techniques and tools. They are also effective communicators who are able to simplify complicated concepts and present them with practical real life examples.

Contact MBA for a course catalog detailing the terms and conditions for training, along with the current prices of all courses.

**GENERAL CONSULTING**

The concept behind MBA Consulting Services is actually quite simple: Provide clients with senior people who have extensive experience in I.S. and equipped with proven methodologies and tools, specifically the "PRIDE" INFORMATION FACTORY.

MBA consultants are not junior programmers nor recent college graduates. They are seasoned professionals with an average of 20 years of experience in I.S. management. They are supported by MBA's "PRIDE" family of products, an integrated product line providing tremendous leverage for analyzing businesses and designing systems. The "PRIDE" methodologies constitute the analysis, planning and development framework with which MBA consultants operate. "PRIDE" is an effective means for managing projects, producing quality deliverables at competitive prices, and assuring customer satisfaction.

The area of expertise for MBA consultants encompasses a wide range of services related to Information Resource Management:

- 1. ENTERPRISE ENGINEERING RELATED ACTIVITIES** - This line of work is concerned with studying a business and formulating an enterprise information strategy synchronized with business objectives. Activities within this area include:

Modeling a Business, logically and physically.

Defining the information required to operate and manage a business.

Documenting a company's systems portfolio and technology portfolio.

Performing an Organization Analysis and Skills Assessment.

Calculating the priorities of business objectives and supporting projects.

- 2. INFORMATION SYSTEMS ENGINEERING RELATED ACTIVITIES** - This area is concerned with the design and development of enterprise-wide information systems, complete with business processes and specifications for software engineering (programming). Activities within this area include:

Performing Feasibility Studies to specify information requirements and devising a system solution (e.g., design a new system, modify an existing system, purchase a package, or combinations).

Evaluating purchased packages.

Designing integrated systems.

Re-Engineering Business Processes.

Documenting Current Systems.

Writing User Manuals and Help Text.

Specifying Software Requirements.

Testing & Implementing Systems.

Auditing Systems Development projects.

Software specifications can be implemented by the client's in-house programming staff or by sub-contractors provided and supervised by MBA consultants.

- 3. DATA BASE ENGINEERING RELATED ACTIVITIES** - This area covers the design and development of the corporate data base so that it is naturally synchronized with applications. Activities include:

Designing/Modeling the Logical Data Base.

Evaluating Data Base Management Systems (DBMS).

Designing the Physical Data Base.

- 4. PROJECT MANAGEMENT RELATED ACTIVITIES** - This includes planning and controlling the human resources required to implement project work. This may include client personnel and/or contractors. Activities include:

Defining Work Breakdown Structures (WBS) and precedent relationships.

Estimating and scheduling projects.

Managing project personnel.

- 5. OTHER CONSULTING SERVICES**

Conducting Computer Technology Appraisals.

OS/2 related computing.

I.S. related research assignments.

## MBA CONSULTANTS

Our most valuable asset is our people. MBA consultants consist of company employees as well as independent contractors skilled in the use of "PRIDE" products. Because of this, MBA consultants are a united group of people who speak a common language and perform well as a team, and not a diverse group of individuals.

MBA consultants have backgrounds in all facets of I.S. management. They understand the problems plaguing I.S. and can call upon their experience to solve problems in a responsive manner. They are not neophytes, but seasoned professionals with impeccable credentials.

The consultants have graduate and undergraduate degrees in Business, Communications, Computer Engineering, Computer Science, Industrial Engineering, Industrial Psychology, Journalism, and Sociology. They actively participate in professional and technical societies and hold various trade related certificates; e.g., CDP, CSP. In addition, MBA actively participates in the IBM Developer Assistance Program (DAP). Our expertise has put us in demand for speeches to prominent business groups and for editorial commentary in industry and trade publications.

*"The company conducts business with many of the most prominent companies from around the globe and brings together professionals from many different geographical locations."*

*"With such skilled workers, the company stays in tune with the products and services it markets and the individual needs of its diverse customer base."*

*"MBA is certainly a leader in its field."*

- DATAPRO

## WHAT'S DIFFERENT?

1. MBA Consultants offer workable solutions described in common business terms. They have a business and management perspective as well as possessing technical skills. Esoteric methods couched in cryptic vocabulary is avoided.
2. MBA Consultants offer depth and experience in a wide range of I.S. related subjects. MBA does not offer inexperienced junior people requiring considerable supervision.

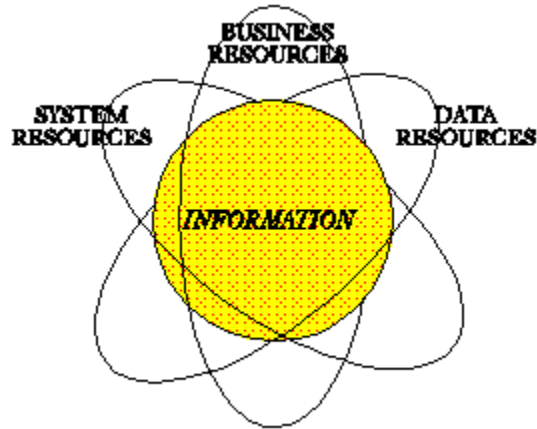
2. MBA Consultants are supported by the "PRIDE" family of products which are demonstrably superior to other products of their kind in the industry. "PRIDE" methodologies, which have been tested by MBA customers around the world, improve communications between MBA consultants and clients, expedite project assignments, and promote quality deliverables.
3. The MBA approach promotes the integration of information resources. Sharing and re-using resources unifies development activities and expedites the implementation of change.
4. Through the "PRIDE" methodologies, MBA Consultants can make a smooth transition from a planning assignment to a design and development assignment.
5. MBA Consultants conduct themselves in a professional manner according to high ethical standards.

MBA is confident that we can offer you superior service at a competitive price. For more information, please see [About MBA](#)

END

COPYRIGHT (C) MBA 1995

**THE IRM REVOLUTION: BLUEPRINT FOR THE 21ST CENTURY**  
THE AUTHORITATIVE GUIDE FOR INFORMATION RESOURCE MANAGEMENT  
(IRM)



**by Milt & Tim Bryce**

MBA Press

ISBN 0-9621189-0-7; 265 pages

Information Resource Management (IRM) represents an imaginative new way of thinking and conducting business in the information age. It is not just another way of using computers. Rather, it is concerned with the development and control of all the resources required to produce information. To many, IRM will represent a radical departure from the mainstream of thinking in today's data processing world which is generally technology oriented. The purpose of this book is to convey to executives a practical approach for the design and development of information resources consistent with their business objectives. Ultimately, the IRM approach demystifies the information systems development process and puts control back in the hands of executive management where it belongs.

**CHAPTERS INCLUDE**

- \* IRM: The Prophecy
- \* IRM: The Concept
- \* The Information Factory: Implementing the Concept
- \* Project Management
- \* Enterprise Engineering
- \* Information Systems Engineering
- \* Data Base Engineering
- \* Glossary of Terms

The concepts and ideas contained in this book represent over thirty years of practical experience in the field and have been proven effective by over 1400 installations around

the world. Since 1976, many prominent Japanese companies have implemented these ideas, including top "Fortune 100" companies and several winners of the Deming Prize for quality.

*"This book belongs in every DP Manager's library."*

- DPMA

*"The information presented in this book reflects clear thinking based on years of practical experience."*

- IRM Association (IRMA)

*"Covers in detail practical concepts and philosophies for utilizing information to increase productivity and competitiveness in an organization."*

- Applied Computer Research (ACR)

**#1 Best Seller in Japan**

The book is priced at \$50 (U.S.). Quantity discounts are available. Consult MBA for deliveries outside of the U.S. and Ontario, Canada. Make checks, money orders or purchase orders payable to:

**M. Bryce & Associates, Inc.**

777 Alderman Road

Palm Harbor, FL 34683

Tel: 813/786-4567

Fax: 813/786-4765

BBS: 813/786-4864

E-Mail: [TimB1557@aol.com](mailto:TimB1557@aol.com)

CompuServe: 76235,2364

Since 1971: *"Software for the finest computer - the Mind"*

**END**

**COPYRIGHT (C) MBA 1995**





