TITLE: "RE-INVENTING BUSINESS PROCESS DESIGN"

by Tim Bryce
Managing Director
M. Bryce & Associates (MBA)
P.O. Box 1637
Palm Harbor, FL 34682-1637
United States
Tel: 727/786-4567
E-Mail: timb001@attglobal.net
WWW: http://www.phmainstreet.com/mba/
Since 1971: *"Software for the finest computer - the Mind"*

"Systems will fail more for the lack of administrative procedures than well written computer procedures."
- Bryce's Law

INTRODUCTION

Back in June 1975 I attended my first systems workshop from Les Matthies, the legendary "Dean of Systems." Seems like yesterday. For those of you who do not remember him, Les came up through what was called the "Systems & Procedures Departments" of the 1940's and 50's. In Les' case, he was recruited by an aircraft manufacturer in the U.S. mid-west during World War II and was charged with establishing procedures for the production of aircraft thereby expediting the development and delivery of planes to the war front. Les was a quick study and was very effective in this regard. So much so, he went on to write numerous books on systems and "playscript" procedure writing. He also conducted courses on systems theory up until his death on December 31st, 1999.

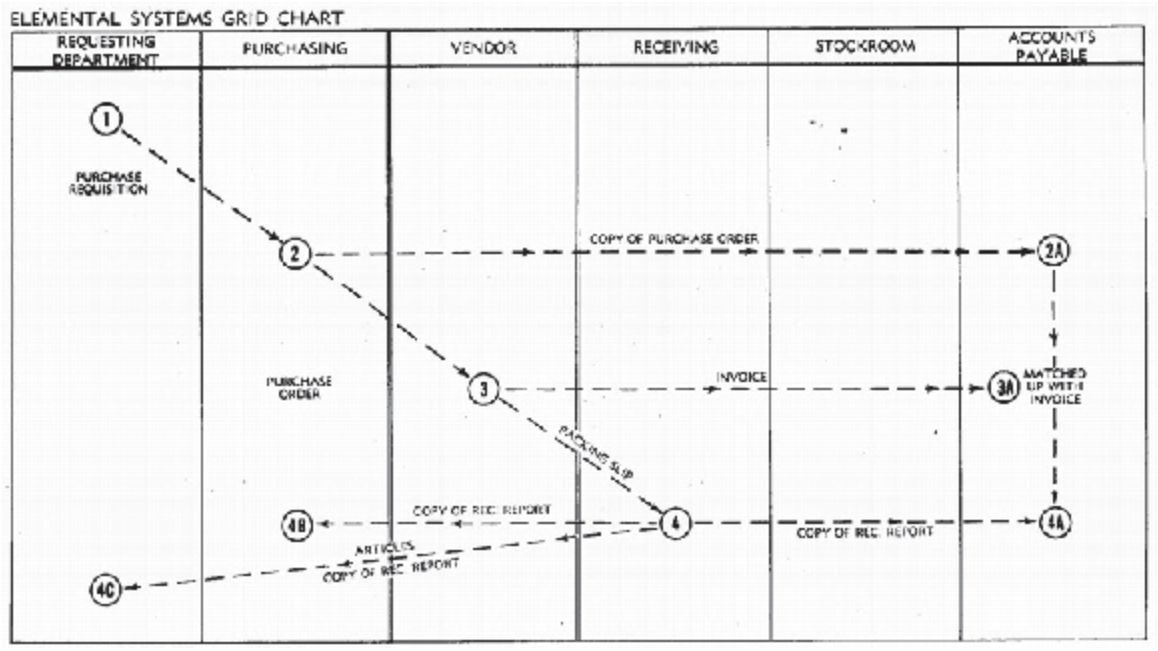
During Les' courses he promoted the use of a simple "Grid Flow Chart" (Figure 1) to track the flow of work between departments. This was a standard technique used for many years in systems departments. As the computer came in vogue, and different program flowcharting techniques were introduced, the Grid Flow Chart was eventually phased out. Regardless of how graphically elegant you think the diagram is or is not, it was a simple and convenient way to express flows of work.

During the 1980's and 1990's, the emphasis was on "structured programming" and then "object oriented programming," and the concept of business process design was forgotten. Basically, the industry shifted its focus from Systems Analysis to Programming. Inevitably, the absence of "work flow analysis" (as it was once called) began to be noticed as software was developed that didn't work in harmony with the business. Consequently, "business process engineering" is being re-discovered by a new generation of developers.

The design of business processes was always an inherent part of the "PRIDE"-Information Systems Engineering Methodology (ISEM) since its inception in 1971. However, we referred to it as "Sub-System Design" (as we still do to this day). In the early days of "PRIDE,"

(continued on page 2)

FIGURE 1



(continued from page 1)

some customers liked to bypass Phase 3 "Sub-System Design" in order to get to the programming phases as soon as possible. Consequently, such customers ran into the problem of developing disjointed software out of step with the business flow. In other words, skipping Phase 3 would inevitably come back to haunt them.

SUB-SYSTEM CONCEPTS

Under "PRIDE"-ISEM, a sub-system is a business process that exists within a unique time frame; e.g., Daily, Weekly, Monthly, Annually, or Upon Request. This timing nuance is a recognition that business processes operate routinely in specific cycles. Further, it is a derivative of the complete specification of information requirements whereby information is needed by users in specific time frames.

For more information on "Defining Information Requirements," see "PRIDE" Special Subjects Bulletin #4 - Dec 27, 2004 at:

http://www.phmainstreet.com/mba/ss041227.pdf

There are three variables pertaining to timing:

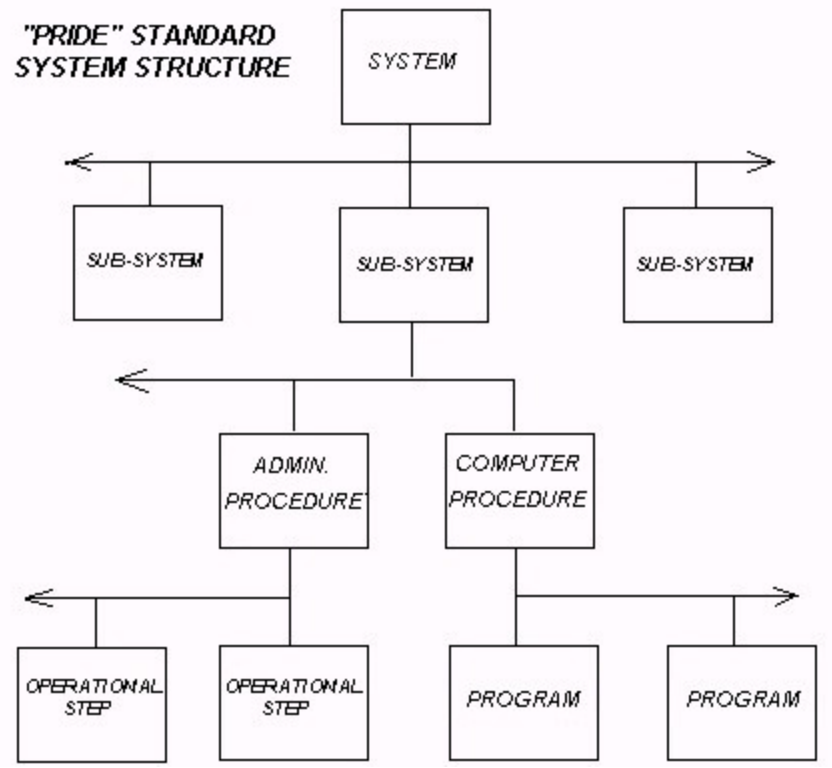
Frequency - specifying how often the cycle must occur; e.g., Daily, Weekly, Monthly, Upon Request, etc.

Offset - specifying when the time cycle is to begin; e.g., 1st of the month, end of the week, etc. As an aside, if the Frequency is 'Upon Request' there is no schedulable Offset (we want the information at any given moment).

Response Time - specifying the maximum amount of time to process the data to produce information; e.g., 5 seconds, 1 hour, 2 days, etc. Note: Response time is NOT a measure of machine throughput, although it will effect computer processing later on.

In "PRIDE"-ISEM we use these timing variables in a technique called "Chronological Decomposition" which is used to collect, store, and retrieve data in a timely manner, thereby keeping the processes synchronized with the data base. These timing parameters will also influence our method of implementation for the sub-systems. For example, if something is desired 'Upon Request' with a five second response time, in all likelihood we are probably

(continued on page 3)



(continued from page 2)

looking at an interactive application. Conversely, a Weekly or Monthly process with a one hour response time might suggest a simple batch process. In other words, timing is a convenient means to define sub-systems and helps determine a suitable implementation of the business process.

There are basically three types of sub-systems: File Maintenance (to collect and store data in a timely manner), Produce Information (to retrieve data in a timely manner), and a combination of both (read/write). As sub-systems are designed, the data is organized into application logical files which are defined in terms of when they are Created, Updated, and Referenced (C/U/R).

The decomposition of a system into its sub-systems is performed in Phase 2 of "PRIDE"-ISEM. At this time, the sub-system is defined only in terms of logically "what" must be processed and "when." It will not be until Phase 3 when we will determine physically "who" and "how" each process will be executed.

For details on "PRIDE"-ISEM Phase 2, see:

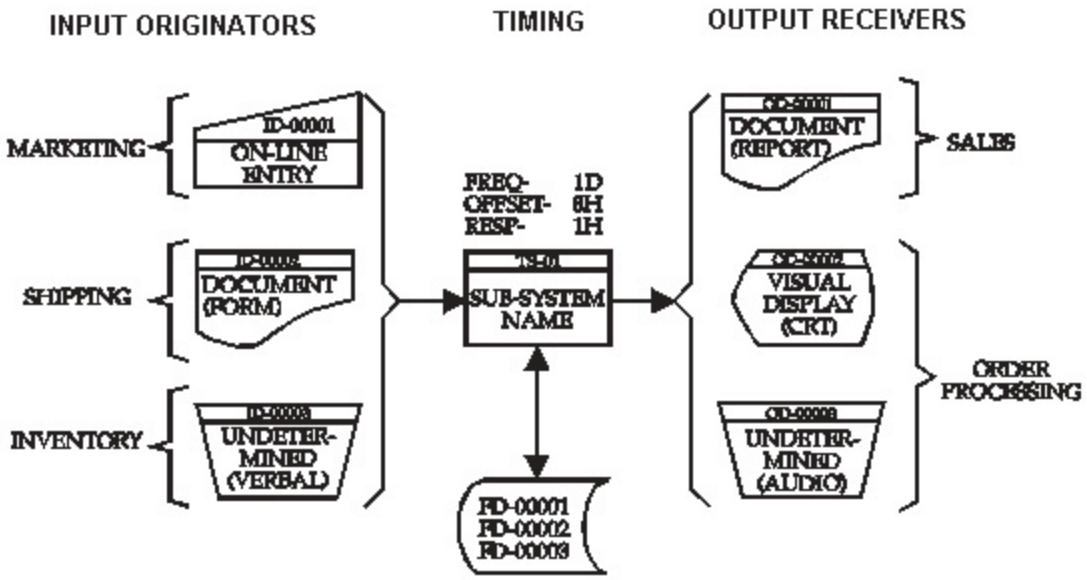
http://www.phmainstreet.com/mba/pride/is20.htm

Following the completion of Phase 2, each sub-system follows its own Phase 3 (Sub-System Design) where it is decomposed into the procedures required to implement the sub-system. Phase 3 is where the "work flow" of the business process is detailed in terms of physically "who" and "how" the process is to occur, from start to end. Here is where we prescribe the use of a "process diagram" to express the business process. Such a diagram (or a "Sub-System Flowchart" as we refer to it) can be drawn either horizontally or vertically depending on preferences. Either way, the diagram describes two things: the flow of work in the sub-system, and; the flow of data in the sub-system.

For the work flow, there are essentially two types of procedures involved: Administrative (the procedures people will follow) and Computer (representing the programs to be executed). Under the rules of "PRIDE"-ISEM, a sub-system can have one or more Administrative Procedures and one or no Computer Procedures (Yes, Virginia, systems can be implemented without the use of the computer). Systems will fail more for the lack of administrative procedures than they will for well written computer procedures.

(continued on page 4)

SAMPLE SYSTEM FLOWCHART
(Showing a single Sub-System)



Phase 2 defines logically "What" and "When."

(continued from page 3)

In laying out the process flow, a line is drawn representing the flow of work with a "Start" to the process and an "End". Following the "Start", procedures are defined based on three constructs:

1. **Sequence** - representing consecutive steps in processing.

2. **Iteration** - representing repetition until a condition is met.

3. **Choice** - representing a selected path based on a prescribed criteria.

This means a process diagram can be drawn as simply or as extensively as desired. For example, it is not inconceivable for a sub-system to have multiple "Starts" and multiple "Ends."

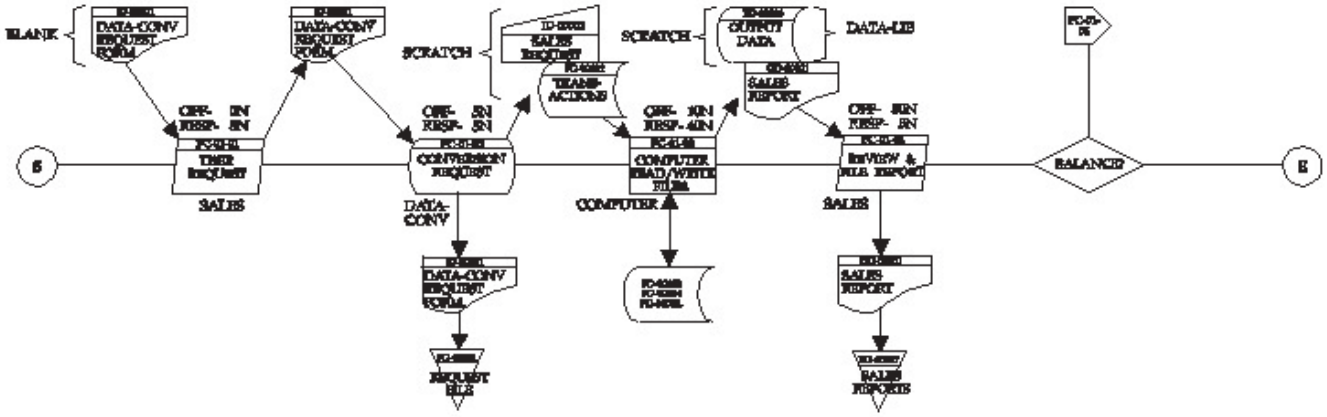
The other aspect of a process diagram is the depiction of the data flow as represented by the inputs, outputs, and files associated with all of the procedures and how they are used (C/U/R). This reinforces a basic "PRIDE" concept: "the only way systems communicate internally or externally to other systems is through shared data."

Below are a couple of simple examples:

(continued on page 5)

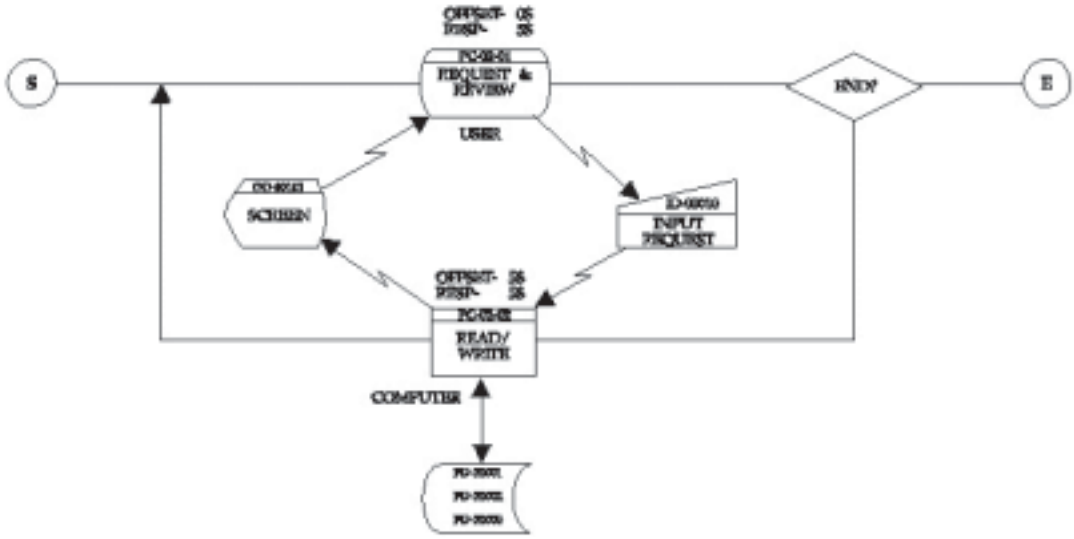
SUB-SYSTEM TIMING
FREQ- R
OFFSET- N/A
RESPONSE- 1H

BATCH EXAMPLE



SUB-SYSTEM TIMING
FREQ - R
OFFSET - N/A
RESPONS - 10S

INTERACTIVE EXAMPLE



(continued from page 4)

Earlier we mentioned a sub-system can have no more than one computer procedure. Let us not forget a computer procedure consists of one or more programs. Normally, there are administrative procedures before and after the execution of the computer procedure. As such, we must remember one characteristic of a sub-system: once a sub-system starts, it continues uninterrupted until its logical conclusion. We have been challenged on this rule time and again by "PRIDE" users. Perhaps the best example is a computer procedure executing routinely on a given cycle (e.g., daily, weekly) with seemingly no human interaction (for example, the computer procedure simply updates or backs-up files and produces no report). However, in this example, there is, in fact, an administrative procedure after all. Care to guess? A simple administrative procedure to trigger or kill the computer procedure. After all, it didn't initially start-up by itself did it?

This "one computer procedure per sub-system" rule has been somewhat controversial over the years, yet we have never seen it fail in 34 years of "PRIDE." It also has an added benefit of providing a convenient means to document our current systems. By scanning our control language libraries (e.g., command files, JCL, etc.) we can detect our computer procedures and thereby deduce our sub-systems.

Regardless of the types of procedures available to us as designers, the Systems Engineer must ultimately determine a practical solution. Since the sub-system must be implemented by human beings (as well as the computer) considerable thought must be put into the sub-system's ease of use ("user friendliness"). Let us not forget an elegant solution that is not easy to understand or use solves nothing.

STANDARD TEMPLATES

Back in 1979 we created an add-on to our "PRIDE" product line with a feature called ADF (Automated Design Facility) which we later renamed ASE (Automated Systems Engineering). ASE implemented the "PRIDE"-ISEM technique of "Chronological Decomposition" and could automatically design systems into sub-systems, procedures (both Administrative and Computer), and programs. ASE was most definitely NOT a program generator, but rather a systems generator. As such, it was a handy precursor for program generators as it would define inputs, outputs, records, and files, and then marry them to the various processes. Regardless, one of the lessons we learned in building ASE was there are some basic sub-system templates covering the majority of all business processes. True, designers can add or eliminate procedures from the ASE sub-system design, but the lion's share of sub-systems used in a business followed the templates.

The point is, a company should develop similar templates for use in designing their business processes. Such templates can save an enormous amount of time during a development project.

CONCLUSION

The design of business processes is hardly a new concept; the need for it has only been re-discovered. However, there are now several interpretations now on the market, some simple, some cryptic. Regardless, business process design represents the missing layer of development that was lost for a period of time. The main benefit of business process design (or sub-system design as we refer to it) is that it ties software engineering efforts with real-world use of systems, thereby making software more usable and minimizes the amount of development time lost on software that will not be used.

Although I find the current business process design renaissance amusing, there is a whole new generation of developers out there who have simply missed it. It is encouraging to see people re-discovering this lost and sorely needed talent. As Les Matthies was fond of saying, *"Systems are for people."* Remarkably, we lost sight of this simple concept. Hopefully, we are regaining our eyesight. I guess what goes around, comes around.

For details on "PRIDE"-ISEM Phase 3, "Sub-System Design," see:

http://www.phmainstreet.com/mba/pride/is30.htm

For more information on "PRIDE" Flowcharting Symbols, see:

http://www.phmainstreet.com/mba/pride/isspfs.htm

For more information on our philosophies of Information Resource Management (IRM), please see the "Introduction" section of "PRIDE" at:

http://www.phmainstreet.com/mba/pride/intro.htm#irm

END

(continued on page 6)

(continued from page 5)

"PRIDE" Special Subject Bulletins can be found at the "PRIDE Methodologies for IRM Discussion Group" at:

http://groups.yahoo.com/group/mbapride/

You are welcome to join this group if you are so inclined.

"PRIDE" is the registered trademark of M. Bryce & Associates (MBA) and can be found on the Internet at:

http://www.phmainstreet.com/mba/pride/pride.htm