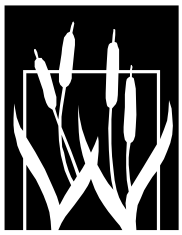

Whitemarsh
Information Systems Corporation

Section of
Migrating Legacy Systems

Whitemarsh Information Systems Corporation
2008 Althea Lane
Bowie, Maryland 20716
Tele: 301-249-1142
Email: mmgorman@wiscorp.com
Web: www.wiscorp.com

1. Introduction

Legacy systems are yesterday's great advances, not just yesterday's leftover baggage. Consequently, legacy systems will constantly be with us. Legacy systems developed over a number of years, continue to evolve today, and will evolve far into the future. To consider otherwise is to chase the future wearing blinders. Current data from the past soon becomes the basis for future trends.

Currently, legacy systems are commonly well over a 1000 programs operating over large, multiple billion character databases. These databases are commonly IMS from IBM. The documentation of legacy systems is fragmented, and of varying qualities. Maintenance of legacy systems becomes more and more difficult with each passing release due to this fragmentation and inconsistency.

If legacy systems were ever able to become complete, this process could ultimately quiesce. However, legacy systems will never quiesce. They must continue to be modified in a timely way to meet changing demands. Changing are their back ends, front ends, sides, and all that comes in between.

Legacy systems, typical of most older data processing systems were not built to be changed. Rather, they were built to be finished. Since however, they cannot be finished, they must be reconfigured so that they can respond to changes in a timely manner.

Enterprise database with the continuous flow cycle (see Figure 9) is both a strategy for reconfiguring legacy systems into changeable systems, and can be the cornerstone needed for evolving legacy systems to meet the demands of client/server.

2. Critical Characteristics of a Legacy System

At the heart of any changeable system is its model. The model must be constructed such that it can be investigated to quickly determine whether a proposed change tactic has a high probability of accomplishing both the change and of not destroying another aspect of the system. The model must be stored in a repository that promotes change investigation and formulation. That is why the repository is the heart of the continuous flow cycle (Figure 9).

Notwithstanding the mechanics of changing a model, the model must lend itself to change. A major characteristic of being able to change is third normal form. This catch phrase has application to both processes and data. A process is in third normal form if it contains no nested processes; if it represents only a single function; and if the single function contains no hidden agenda alternatives. A data structure is in third normal form if it does not contain repeating groups or vectors; and, if all the data elements are fully dependent on the complete primary key of the structure; and finally, if no nonprimary key data element within the structure depends on another nonprimary key data element for its full meaning.

Notwithstanding the third normal form nature of data and processes, the model must clearly fulfill the business' missions. Determining the complete set of business missions requires a top-down policy analysis.

Most legacy systems are not repository based. Thus, they cannot be changed easily or with any confidence that when a problem is fixed or when an evolution is accomplished such that latent failures have also not been installed.

The first step then for legacy systems migration into Enterprise database is to reverse engineer their specifications into the repository. Only when this is accomplished will the repository's report writer be able to aid in the identification of common, redundant, or conflicting data, systems, processes, or procedures.

3. Legacy Systems Redeployment Alternatives

There are three alternatives for reworking legacy systems into a changeable system. These are:

- Big bang
- Inductive
- Plan then parts