TITLE: "CREATING UNIVERSAL SYSTEMS"

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"*

"There is only one problem with common sense; it's not very common."
- Bryce's Law

GENERAL DISCUSSION

In this day and age of "globalization" more and more Information Systems are crossing geographical boundaries. Because of this, serious consideration should be given to making systems universally applicable to any country. Some might consider this an impossible task, but it is actually easier than you might think. It just requires a little common sense and some planning.

The biggest problem in making universal systems is that programmers tend to bury too many of the details of a system down in the program source code, which is not a good place to tinker around in. Instead, certain elements of the system should be placed in separate files thereby making it convenient to translate. Consideration should be given to creating separate files for:

* **PRINT MAPS** - An output, such as a report or printout, can be decomposed into various sections (and represented in the IRM by the RD resource). When a program is executed, one of the parameters should be the desired language (e.g., English, Spanish, German, French, Japanese, etc.). Based on this parameter, pertinent print maps are called from the "Print Map File" to assemble the requested output.

* **SCREEN PANELS** - This is similar to the "Print Map File" whereby the sections or a screen can be decomposed into its various panels (again using the RD). As a program is executed, pertinent panels are called from the "Panel File" to build the screen.

* **MESSAGES** - Messages are too often buried in source code. Instead, they should be placed in a separate file for printing or display in a screen. The individual message is also recorded as an RD in the IRM Repository.

* **HELP TEXT** - Help text should also be maintained separately for easy retrieval. Again the RD is used to represent Help text.

Separating Maps, Panels, Messages, and Help text from program source code, makes it easy to translate to foreign languages. Further, it encourages developers to share and re-use resources, thereby contributing to integrated systems.

A serious consideration in the Far-East is the **Double Byte Character Set** or **DBCS** which is used to accommodate Japanese and Chinese Character alphabets with voluminous characters. To construct one such character, two bytes must be stored in a single byte (hence the name "DBCS"). Fortunately, the technology has evolved and DBCS is implemented in most operating systems today. However, developers should be cognizant of this requirement, particularly as they are designing Inputs, Outputs, and Files. Check with your hardware or operating system vendors for specifics. Better yet, check it out on the Internet.

INPUT/OUTPUT DESIGN

During design of the Inputs and Outputs in "PRIDE"-ISEM Phase 2, consideration should be given to the expression of certain types of data elements; for example:

* **DATES** - How dates are to be expressed may vary from country to country; for example: Nov 13, 2004 - 13 Nov, 2004 - 2004-11-13. How a date is presented to an end-user is different than how it is physically stored.

* **TIME** - This is similar to dates; some people like to see AM/PM, others like military time, e.g., 14:30 (2:30pm)

NOTE: Regardless of how Dates and Times are to be physically presented to the user, standards should exist to express how dates are to be physically stored, such as "YYYYMMDDHHMMSS" (Year/Month/Day/Hour/Minute/Second). Failure to do so caused the horrendous Year 2000 (Y2K) problem a few years ago.

* **TIME ZONE** - Representing local time.

* **CURRENCY** - What form of monetary values should be expressed; Dollars, Yen, Marks, Pounds, Euro Dollars?

* **MEASUREMENTS** - Accommodate different units of measures for weights (pounds vs. grams), distances

(continued on page 2)

(continued from page 1)

(miles vs. meters), and temperatures (fahrenheit vs. centigrade).

* **TEXT** - The Western world prefers viewing text horizontally from left-to-right, but as we go into the Eastern countries, they like to see text vertically, sometimes right-to-left.

Many operating systems today provide the means to capture such settings. However, it might be necessary to establish a separate "Personal Settings File" for a particular Information System.

Attention should also be given to DEFAULT settings, particularly at time of input. Further, where applicable, consider auto "UPSHIFTING" or "downshifting" text as needed. For example, most Internet addresses (such as a URL or e-mail address) should be downshifted.

The techniques mentioned above are simple and effective to implement. It is important that a translation strategy be considered as part of the system design. During design, your mantra should be *"Know your audience; make it usable; think Global."*

NOTE: This paper is excerpted from the "PRIDE" Methodologies for IRM at:
http://www.phmainstreet.com/mba/pride/pride.htm

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

END

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

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