SynCo Technologies Inc.
124 West 24th Street Suite 2b
New York, NY 10011

Generic Design
Position Statement

Philosophy

As the software industry has matured it is apparent that the most costly resource of an application life cycle is programmer time, both for development and on-going system maintenance. SynCo Technologies, Inc. (formerly Synectic Software, Inc.) is dedicated to the concept of generic, rule-based programming and design as a strategy to reduce these costs by moving the application design process to the business people.

SynCo believes that the effort it takes to put together software in which the logic is solidly and firmly programmed, but specifics are stored external to the code itself provides a better return on investment than hard-coded, "one-off" systems.

As indicated in the Constructive Cost Model (COCOMO), published in Dr. Barry Boehm's book *Software Engineering Economics*, Prentice Hall(1981) and updated in 1999 as COCOMO II, the programming resources required in the software development cycle ranges from 40 – 70 percent of the total resources of the project. This percent includes creating detail design specifications, transferring business knowledge to the technical staff, coding, and unit testing of the application. If this analysis is expanded to include the entire software life-cycle these percentages would persist or increase for the on-going maintenance required.

Historically, one attempt at an automated solution to reduce costs in maintaining applications and incorporating design changes was Computer Aided Software Engineering (CASE) tools. CASE tools incorporate the definition documentation into the application design. Unfortunately, when business rules change and definition documents are modified, the application must then be recompiled and redeployed. SynCo's philosophy of generic systems moves the maintenance of the business rules, presentation and data structures to the application itself.

These "generic" systems tend to have longer life cycles due to reduced maintenance and greater user empowerment since the control of the business logic and presentation is in the hands of the users. Benefits are realized when business rules change either frequently or significantly months or years after initial implementation. Often software may be difficult to maintain by a programmer coming to the project late when analysts and testers are no longer on the scene. The logic of a generic system has already been tested and, ideally, should need no re-programming and re-testing when a business rule or user need changes. Even when a major requirement changes and re-programming is necessary a generically programmed system is preferable. Since one of the tenets of generic systems is maximum re-use, there is less code than in a hard-coded system. The code is cleaner, clearer and easier to maintain. When programming changes are made, the changes add rule-processing capabilities that enhance the software in ways that carry into the future instead of simply addressing a current need.

SynCo Technologies Inc.
124 West 24th Street Suite 2b
New York, NY 10011

Experience

SynCo has used its philosophy in its commercial product MediEase, which tracks immunization events and calculates immunization compliance based on flexible data based rules. Immunization rules are quite complex. For example, series of vaccinations often have different schedules based on the age of the patient, the time span between 2nd and 3rd shots, condition/drug interaction, etc. Because of its flexibility MediEase may be used with any state's immunization rules and any health professional requirements that may come up. MediEase handles new immunizations and new rules for immunizations without intervention by programmers. Without requiring new releases, MediEase now tracks new immunizations such as Lyme disease and has responded immediately to changes in the CDC recommendations for childhood immunizations. Users have added their own immunizations; adjusted the rules and added non-immunization events that the designers of MediEase could not have predicted.

The magic of the generic programming concept as applied in MediEase is that if one immunization works properly, then all current and future immunizations work properly. MediEase provides compliance tracking to over 350,000 people in over twenty universities nationwide with a diverse set of contraindication and immunization rules.

SynCo has used its philosophy in its front-end payroll system CAPS (Compensation and Payroll System). CAPS was first used at a large Wall Street financial institution with complex payroll requirements. The definition of data elements that determine how employees are paid, how deductions are taken and which benefits are granted are all data based. When payroll rules change, and they often do, and new data elements are required, they are simply defined in CAPS' metadata repository. Payroll has over 40 import and exports in various formats from multiple sources. In order to accommodate this range of interfaces, SynCo developed a generic interface module. Interface templates and rules are defined dynamically. In conjunction with the metadata, interface data elements are mapped to the appropriate CAPS data field and tables. As a result, many costly hours of programming time have been saved. Whenever a change to an interface structure occurs, a simple user modification to the template is all that is necessary.

Vision

The next evolution in the concept of generic rule-based programming is the generic application development model.

This model transfers the implementation of an application's presentation, navigation, workflow, and business rule logic from the programmers' hands to the domain of the analyst or subject matter expert (SME) thereby empowering them to present the system as envisioned.

The generic application development model moves the component object reuse paradigm from the arcane world of the programmers to the more familiar, accessible world of the business analyst and then raises its power to the nth degree. Integrate this concept with the immediate worldwide deployment inherent with the infrastructure of the Internet and you have a software tool of unprecedented power.

SynCo is committed to this philosophy and is currently developing Gen-I-Sys (Generic Internet Systems). Gen-I-Sys is the culmination of our experience of designing and implementing rule based applications. Gen-I-Sys is an all-purpose Internet application that uses a designer to create and maintain data, presentation and workflow definitions to create, on the fly, zero deployment Internet applications.

It will include an enhanced version of the interface module enabling the use of more complex rules and automated processing. Gen-I-Sys will provide an end-to-end application development solution.