Chapter 13
Object Model Development/Engineering: Proven New Approach to Enterprise Software Building

David Chassels
Procession PLC, UK

ABSTRACT
Solving the business “software problem” of inflexibility with poor user experience was at the heart of the original R&D that started over 20 years ago. Any solution had to acknowledge how people work and most importantly remove the interpretation gap between users and “IT,” thus removing the need for programmers in the build process. It was quickly recognised that in reality business logic does not really change, and it was therefore important to separate from the ever-changing technologies, such as operating systems and browsers, to “deliver.” After 20+ years of research and development and working with early adopters, this approach has achieved the objectives set. As is explained, generic task objects and the important links were built and displayed in a Graphical Model where the building of custom applications takes place with no change to the core code and no code generation or compiling. This approach has opened a new perspective on capability in software yet does it by adopting simplicity in the core design. Indeed the driving philosophy for all development was to produce enabling technology that was business friendly, simple to use, and generic in application.

INTRODUCTION AND BACKGROUND
The thinking behind this development has been driven by business thinking with a “clean sheet.” All the developer coders were selected direct from universities and colleges none of whom where “contaminated” by existing commercial development practices. This “journey” started out in the late 1980s at a time many were expecting the removal of coders as articulated by Naomi Bloom (2013):

DOI: 10.4018/978-1-4666-4494-6.ch013
Writing less code to achieve great business applications was my focus in that 1984 article, and it remains so today. Being able to do this is critical if we're going to realize the full potential of information technology.

...how those models can become applications without any code being written or even generated.

The founder was an international banker who shared these views from the perspective of a user of information. He was a business executive but understood enough of “IT” to impose his business ideas into finding a solution. The core thinking was based upon once only entry of data to flow in a horizontal manner (as opposed to vertical silos) addressing all business requirements yet remove the need to use coders to build working applications.

It was established that people at work need to be supported by “generic” task types and this simple reality opened the door on removing the need to program code applications. The early research was carried out using the dos operating system and the Paradox database to create a complete data centric environment utilising the latent power of a relational database. Against then current thinking, it was proven that the identified generic task types and required links could be stored as data in the database ready to be configured.

By 1995, the concept was proven, and it was decided to re-write using Windows 95 and the Oracle database was chosen. Once this new version was created it was decided to create a Graphical Process Designer as an interface over the code that would allow easy click drag and open the task types to allow configuration by business analysts to build the required application.

What follows explains the detail core thinking and how build takes place.

KEY ISSUES

Before a sustainable approach was adopted it was important to understand some basic fundamentals on how organisations actually work.

1. First people are the source creators of all information and second there is the need to recognise that people work in relatively small teams to achieve individual and collective outcomes that make any business or indeed an efficient government. Worthy of note is that book keeping is just that it records history and generally keeps the accountant happy; it does not “run” the business.

2. The other key issue that was established early on was that business logic has never changed since commerce started; indeed business is actually quite “simple” if you focus on supporting people at work. This includes “rules” simple or complex (e.g. means testing) and just reflects business logic. The communication technologies to deliver are both complex and challenging but do not change the fundamentals of business; it is about people, internal and external to the organisation and IT should there to support and contribute to efficiency.

3. When you look at how people work irrespective of the required function there are relatively few work task types, human and system, including the user interface that address all business driven issues? So why repeatedly recode for every function in a business when such standardisation allows unlimited flexibility to build on any business requirement?

With these basic fundamentals and recognising the latent power in relational database technology...