Journal of International Information Management

Volume 3
Issue 3 *Special Edition*

Article 8

1994

A formalized object oriented user driven approach using business forms

Ric Jentzsch
University of Canberra

A formalized object oriented user driven approach using business forms

Ric Jentzsch
University of Canberra

ABSTRACT

Although object-orientation exhibits many promises, there exists many fundamental areas that need to be addressed. Two of these areas are the widespread acceptance of the object mindset, and interactive user involvement in the object oriented business system development process. First, the mindset must grow and be extended not only by the engineering industry but also by the ever expanding service industry. Second, effective, efficient, flexible user driven tools must be developed, in the form of development approaches to reduce the reliance on scarce human resources. In order to accomplish both areas, the technology user is the key element.

This research in progress concentrates on the development of a formalized business form user-driven approach using an object oriented analysis view. The resulting analysis produces an initial conceptual object model that is used by the users and designers for further object analysis, object design, design evolution, and object oriented application domain identification.

INTRODUCTION

The human systems for business and technology are currently at polar opposites. The social-technical relationship that should exist between the two cultures is at arm's length. The use of emerging technologies in today's organizations is stretching that distance even further. To close the gap we must begin by developing cooperative development approaches between users and their business information systems. These development approaches must be actively driven as a cooperative venture between users and technology.

This research is directed towards integrating a common user interface, the object oriented analysis (OOA) phase, an object oriented view, and the nontechnical user. It is in the OOA phase that successful user interaction, coupled with the user task environment and object oriented technology, must produce a better base for a business information system. Object orientation is an emerging technology. As object oriented development approaches continue to mature, lessons learned from conventional development approaches need to be remembered. Conventional development problems should not be propagated to object oriented or other emerging technology approaches. However, far too often this is not the case.

The users have the knowledge, and assisted by the use of technology they can create initial blueprints for their business solutions. The solutions should be from the user's perspective and in terms of the user's task environment. Users can create the solutions using the view(s) that they are most familiar and comfortable with, and these views can be represented in a conceptual object model. This paper describes, from a broad perspective, what is needed for the development of a form based user driven approach. The details of the approach and the knowledge bases are presented in a separate paper currently being finalized by the author.

APPROACH - METHOD AND OBJECTIVE

This research seeks to reduce some of the effects that have plagued both conventional and object oriented development approaches. In order to do this, this research concentrates on the formalization of a user driven business forms view modelling approach within the object oriented paradigm. Each of the four areas identified in Figure 2 represent the boundary and scope for this research.

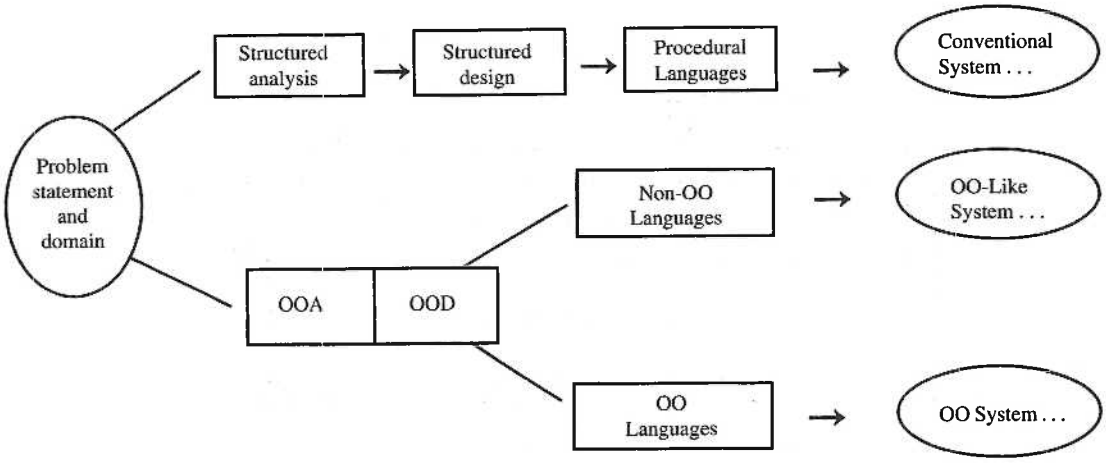
The first step in this process is the development and formalization of a user driven development approach. A user formalized approach must address the problem statement, reflect the user's real world problem domain, and capture the user's knowledge in terms of the user's view and terminology. The best user instrument to capture and present this information is the business form.

The development of an information system begins with a problem statement and its domain. From here the system can be developed using a conventional approach or an object oriented approach (Figure 1).

The business forms approach formalizes the development of a conceptual object model based on analyzing the user's business form(s) and its components. The business form driven approach has been looked at from a few perspectives but not within the object oriented paradigm. The advantages of using the business form analysis approach has been well documented (Choobineh, et al., 1988; Jentzsch, 1992; Lefkovits, et al., 1979; Storey & Goldstein, 1988). Business forms represent one of the most common user views and user interfaces into business and automated systems. These systems can be object oriented, object-like, or conventional business information systems. An overview of where object orientation and OOA fit into the business system design process is presented in Figure 1.

The goal of any design approach is to support applications that the end user finds natural, easy to use, and that factually represent the user's task environment (Kroenke, 1992). The goal can only be met by establishing a cooperative design approach between users and the technology used in developing their business information systems. However, the use of technology as a cooperative venture with users, in developing real world solutions to real world problems, has occurred with varying degrees of success (Bennett, 1987). For both conventional and emerging technologies, more and more technology is being applied to designing the various aspects of an automated system. However, there still exists a great deal of distance between the user and the designing of the business information system.

Figure 1



SCOPE

This research is concerned with developing a real world model using an OOA view within that world. This is done from the perspective of the three entities in Figure 2.

The first two entities:

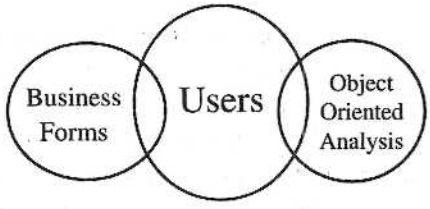
- Users
- Business forms

work together in the first phase of the object oriented development process:

- Object Oriented Analysis

to produce a graphical real world user requirements model referred to as a:

- Conceptual Object Model.

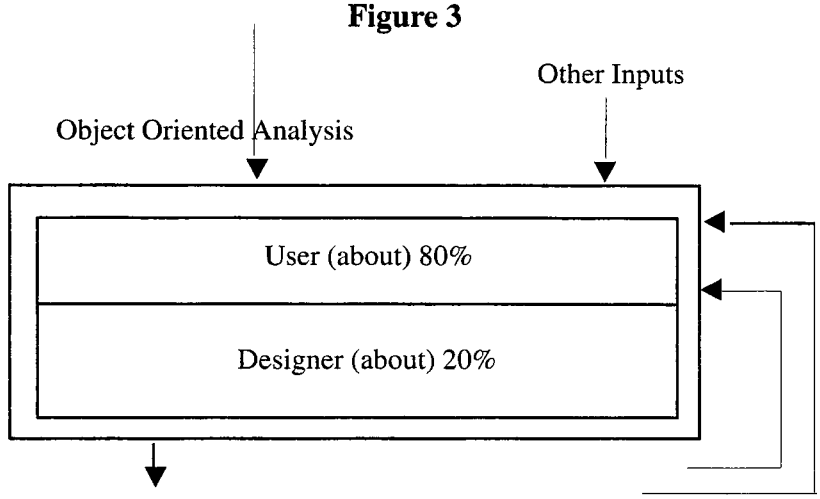


Conceptual Object Model

Figure 2

It is not the intention of this research to create a solve-all *"silver bullet"* approach. This research does not suggest that the object oriented professional designer be excluded from the OOA phase. As can be seen in Figure 3, this research attempts to drive the OOA from the requirements defined by the users. The object oriented designer plays an active role once the initial conceptual object model has been created. Although this research aims for a 80/20 participation split - even a 50/50 participation will solve some of the user-designer-design problems occurring today in the object oriented development process.

Figure 3



Although this research is based on the business input form, other types of forms such as video, mail, audio and even virtual reality could equally apply to user form driven approach.

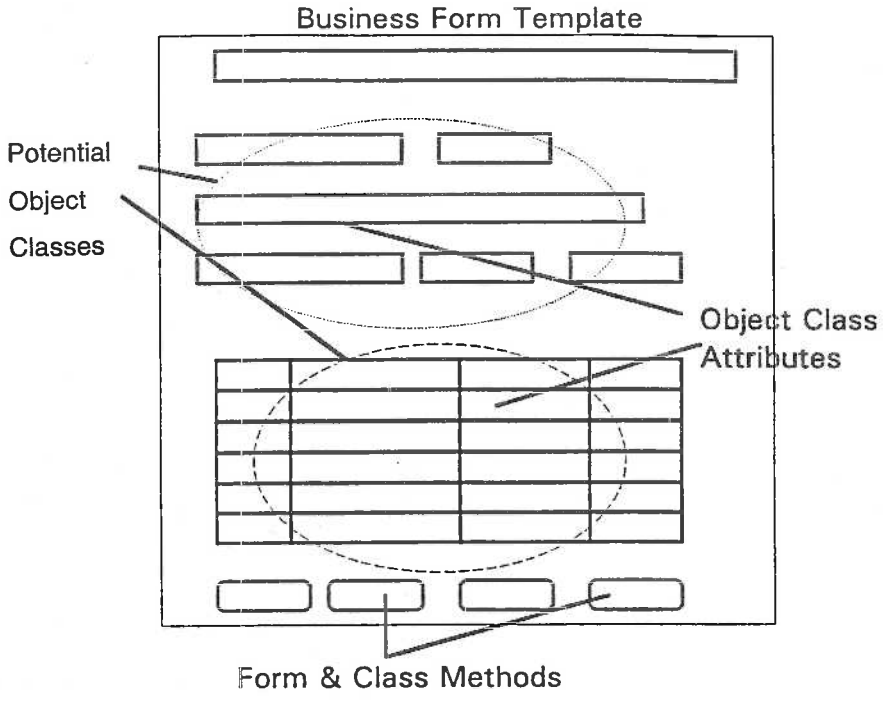
BUSINESS INPUT FORM

The primary instrument in this approach is the user's business input form. Business forms are a formal means of communication. These forms can be the two dimensional paper form or a business form found on a computer screen. These forms document a definable communications interface structure. The business form is the user's view into their business information system and the vehicle in which the business information system interacts with the user's task environment. The form is the most common user interface used in and out of business today.

Users tend to view their functional and behavioral tasks in terms of input and output. The output might be a query display or a hard copy output. Creation, modification, and deletion of information are viewed in terms of input. Such inputs and outputs form the functional and behavioral currency that users work with in their daily, weekly, monthly, and annual processing. It is the structure and behavioral consistency of using forms that make them an excellent candidate to be used in an information system design process.

The business input form can be an external or an internal form. External forms are comprised of such sections as global titles, form titles, and field titles. Internal forms have sections that include such areas as form titles, field titles, fields, routing, processing, automation, instructions, and control information (Tseng & Mannino, 1988). Global titles, in relationship to internal forms, tend to be optional, or organizational area specific and are less common. Figure 4 shows the sections of the business input form in object oriented terms.

Figure 4



OBJECT ORIENTED ANALYSIS KNOWLEDGE BASES

The business forms approach, using object orientation, proposes an interactive user driven approach. The business forms approach interactively generates an initial conceptual object model utilizing the properties as described on the business form. The underlying technology is responsible for establishing an efficient structure while the user is responsible for specifying their problem domain using their business input form. For this to be an effective user driven approach requires the assistance of dynamic knowledge bases.

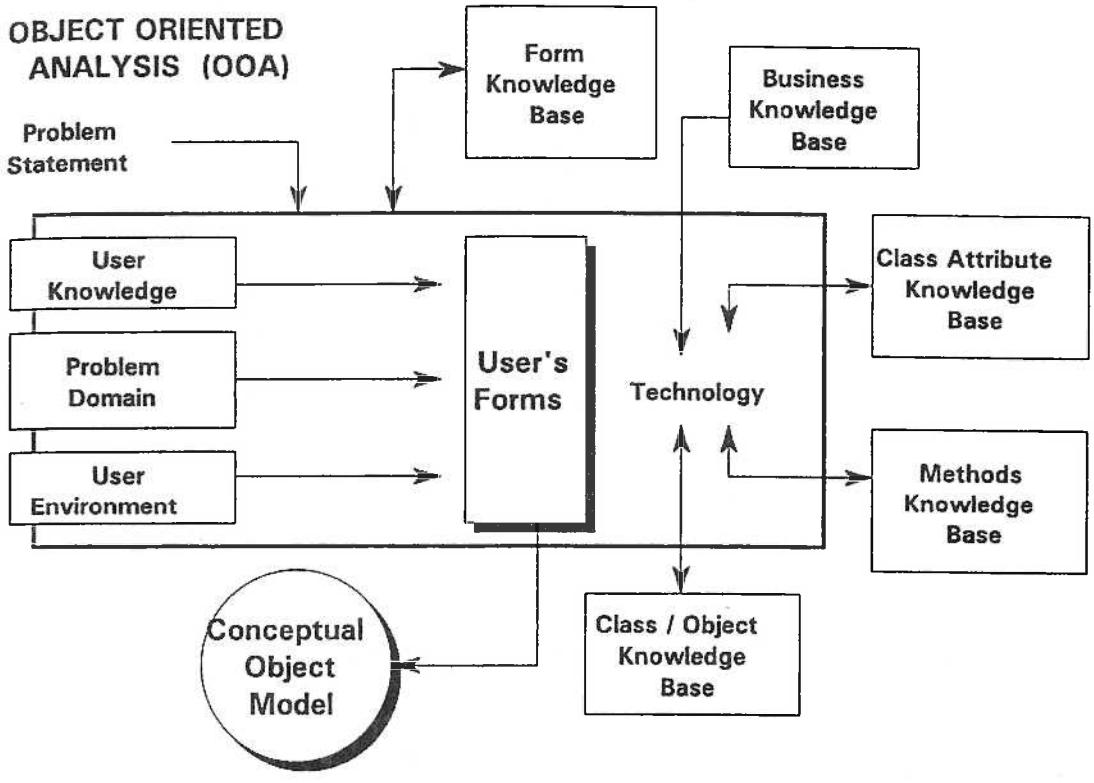
There are five knowledge bases used in this OOA approach. These knowledge bases are: the business knowledge base, form knowledge base, methods knowledge base, class attribute knowledge base, and class-object knowledge base. This approach targets only a portion of the OOA phase (80/20) (see Figure 3). Figure 5 is a detailed view of the OOA phase as defined in this research.

Business Knowledge Base

There are certain rules and constraints that a designer, or the organization, might need to impose on the structure to be developed. This knowledge base provides a means by which this information can be stored and retrieved. This knowledge base would be the responsibility of

someone with an in-depth knowledge of the business environment and its operations. For this research, that knowledge base resides with the business and not the object oriented professional designer or technology user.

Figure 5



Form Knowledge Base

This knowledge base stores information about the form and its image. The form knowledge base is used to identify similar or identical forms. This component interacts with the class-object and class attribute knowledge bases in establishing a predictability index of whether a form is similar, identical, or sufficiently different from previously stored forms. Given the interactions with the other knowledge bases, by the time you come to store the form it is very unlikely that replication will be a problem.

Class Attribute Knowledge Base

Class attributes are created, used, and reused by different forms. This knowledge base is responsible for storing new, and making available previously defined, class attributes. It is left to the user to define the semantics differences between stored and newly created class attributes. Semantically two terms might mean the same thing. It is the user's responsibility to determine if that is the case. Since this approach is user driven, the user must resolve, at least, the initial data semantics. In the OOA phase, after the initial design is completed, the professional designer with the user is responsible for resolving possible semantic differences (Figure 3).

Class-Object Knowledge Base

There are two parts to this knowledge base: existing objects and defined classes. The class-object knowledge base is responsible for storing the details about the classes for object instances. These details are used (with the class attribute knowledge base) in deriving the static dimension for the conceptual object model. The class-object knowledge base is also responsible for letting the user know about previously stored classes. This prevents class duplication and makes class reuse possible.

There are objects that exist and new ones created that get used and reused on business forms. This knowledge base is also responsible for maintaining existing objects, and storing new objects in the form of data types. Data types include the standard programming language data types (string, integer, etc.), derived data types (computational), abstract data types (date, time, user defined) and user defined data types (Booch, 1991). As new data types are created and extended, this component records them for use by subsequent user views. This allows the data types to be reusable.

The class attribute and class-object knowledge base work together. This allows for classes to be semantically different even though constructed from similar or identical class attributes.

Methods Knowledge Base

The methods knowledge base is used to store the behavioral components for the forms, classes, and class attributes. Methods are operations and functions for forms and object-classes. These methods along with user defined methods are stored in this method knowledge base. The methods knowledge base is responsible for making existing and user defined methods available during the form design and description process.

SUMMARY

Overall, there has been no research combining the forms driven approach with object oriented concepts. Furthermore, there has been little to no research on direct user involvement in the phases of the object oriented development process. With the process of deriving an object oriented design from the early analysis not yet being formalized (Alvagar & Periyasumy, 1992) with tool support in the early stages of development (deChampeau & Faure, 1992), and research into user involvement in the object oriented design process just beginning, research and development into this area needs to be expanded.

As early as 1988 both Kroenke and Dolan (1988) discussed the many promising characteristics of object oriented concepts. Now as object oriented technologies are beginning to mature, the opportunities to solve real world user problems has never been greater. Yet, ultimately it is left to the user to define their objects and they must do this in a professional designer environment. Now it is time to begin to apply those promises, to allow users to define their own objects and use the concepts of modern computing with real world active user participation.

REFERENCES

Alvagar, V. S. & Periyasumy, K. (1992). A methodology for deriving an object-oriented design from functional specifications. *Software Engineering Journal*, 247-263.

Bennett, A. (1987, February). Developments in the user and designer interfaces to databases. *ACS Newsletter.*

Booch, G. (1991). *Object-oriented design with applications.* Redwood City, California: The Benjamin/Cummings Publishing Company Inc.

Choobineh, J., Mannino, M. V., Nunamaker, J. F., & Nonsynski, B. R. (1988, February). An expert database design system based on analysis of forms. *IEEE Transactions on Software Engineering, 14*(2), 242-53.

deChampeau, D. & Faure, P. (1992, March/April). A comparative study of object-oriented analysis methods. *Journal of Object Oriented Programming, 5*(1), 21-33.

Jentzsch, R. (1992). View modelling using object oriented concepts. *Proceedings of the Second Software Engineering Research Forum.* Melbourne, Florida, 33-39.

Kroenke, D. M. & Dolan, K. A. (1988). *Database processing*, 3rd Ed. Chicago, IL: Science Research Association.

Kroenke, D. M. (1992). *Database processing.* 4th Edition. Singapore: Macmillan Publishing Company.

Lefkovits, H. C., et al. (1979, February). A status report on the activities of the CODASYL E.U.F.C. *SIGMOD.* Record 10(123).

Storey, V. & Goldstein, R. C. (1988). A methodology for creating user views in database design. *ACM Transactions on Database Systems, 13*(3).

Tseng, V. P. & Mannino, M. V. (1988, November 16-18). Inferring database requirements from examples in forms. *Proceedings of the Seventh International Conference on Entity Relationship Approach*, Roma, Italy.