B usiness
O bject
R eference
O ntology
Program

simplifying semantics

Working Paper

AS2

The BORO Approach: Strategy-2

Using Objects to Reflect the Business Accurately

Issue: Version - 4.01 - 01-July-2001

Contact

For queries regarding this document, or the BORO Program in general, please use the following email address:

contact@BOROProgram.org

AS2

THE BORO APPROACH: STRATEGY - 2

USING OBJECTS TO REFLECT THE BUSINESS ACCURATELY

CONTENTS

CONTENTS
AS2

AS2

THE BORO APPROACH: STRATEGY - 2

USING OBJECTS TO REFLECT THE BUSINESS ACCURATELY

1 A core issue for business objects

Underlying the BORO approach is a core issue for objects and business modelling—how can we accurately reflect the business in a model and thus in a computer system? This paper provides a context to, an outline of, the way in which BORO resolves this issue.

1.1 O-O's original claim

In the late 1980s, object-oriented (O-O) system building became popular. At that time, one of the common claims of the experts was that the objects in their models directly reflected reality. For instance, Ivar Jacobson (in *Object Oriented Software Engineering*, Addison Wesley, 1994) wrote:

> *A model which is designed using an object-oriented technology is often easier to understand, as it can be directly related to reality . . . Since objects from reality are directly mapped into objects in the model, the semantic gap is minimised.*

Peter Coad and Ed Yourdin (in *Object-Oriented Analysis*, Yourdin Press, 1991) gave the same message:

> *OOA directly maps problem domain and system responsibility directly into a model. Instead of an indirect mapping . . . the mapping is direct, from the problem domain to the model.*

It was then generally accepted that the objects in O-O models map directly onto objects in the business. This appeared to neatly resolve the questionof how to produce a model that reflected the businesssufficiently accurately.

1.2 Questioning the original claim

Subsequently, this direct mapping claim has been questioned, at least for the objects in object-oriented programming languages (OOPL). As people gained more experience with these languages, they realised that even though its objects may be better at reflecting reality, they do not do so directly. For example, Steve Cook and John Daniels (in an article entitled "Object-Oriented Methods and the Great Object Myth") wrote:

> *Many authors . . . propose, as though it were obviously the case, that the real world consists of encapsulated resources and predefined access procedures. So we find it stated that a real aircraft has take-off and fly operations, a real cup provides a drink operation, and so on.*
>
> *This view of the world—which we shall call the object myth—is nonsense. If you drink a cup of tea, you do not invoke the drink operation on the cup anymore than the cup invokes the drink operation on your lips, or, indeed, anything invokes an operation on anything else.*

Clearly, the OOPL (and the analysis and design methods based upon them) that lead to things such as cup objects that invoke drink operations do not directly reflect reality. However, this does not mean that the experts' original insight was misguided. We can still use objects to help us reflect reality directly. But to do this, the whole approach needs to be changed. We need to examine what objects in the business are instead of dabbling with the 'objects' in models and programming languages.

1.3 Overturning a business object myth

There is one popular misconceptions that has bedevilled the understanding of what objects in the business are. It has been assumed, even by the experts, that people tend to see the world in terms of objects. For example, Ivar Jacobson wrote:

> *People regard their environment in terms of objects. Therefore it is simple to think in the same way when it comes to designing a model.*

This is profoundly wrong. It is a myth that people currently see or think in terms of objects. This myth has seriously hindered the development of business objects.

Currently, people naturally see the world in terms of attributes belonging to entities (though they might not call them that). When most people see a red car, they think they see a car with the property (attribute) of redness. They are not seeing objects because neither the car nor its red attribute are objects. We examine what they do see in more detail in *OP1—Entity Ontology Paradigm* and *OP2—Substance Ontology Paradigm*.

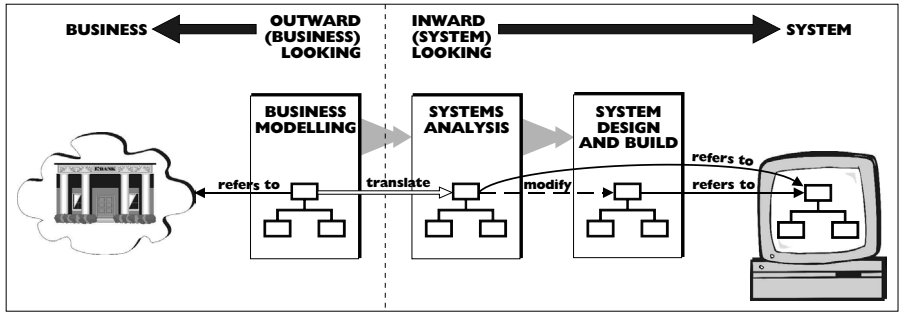
If we want to persuade people that objects are easy to use, then it might seem a good idea to suggest that O-O is based on the way people see the world. If we want our system building to be successful, then it is a terrible idea. It leads us in completely the wrong direction. It stops us from recognising that we need to shift from our current entity (and attribute) way of seeing things to an object way.

1.4 The nature of business modelling

A fundamental reason for these misconceptions about what people see has not been clearly examined. In terms of the traditional stages of system building, we examine the business at the initial 'business modelling' stage (shown in *Figure*

AS2-1). This is the stage at which we should map business objects directly into a model. This is where we should shift to an object way of seeing the business.

Figure AS2-1 Traditional stages of system development



The misconceptions persist because system builders typically think of the business modelling stage in the same terms as the other stages in system building rather than thinking of business modelling in its own terms.

Figure AS2-1 shows that this involves looking outward at the business rather than, as the other stages do, looking inwards at the final system. We also need to recognise that:

- Business models explain what the business does, and
- System models explain how the system will operate.

In other words, the business model works at an understanding level and the system model at an operational level. This is reflected in their objectives. Business modelling's objective is, or should be, to capture an understanding of the business. The later stages use this understanding, but their objectives are aligned with the successful operation of the implemented system. If you work through The BORO Working Papers, you will see many examples illustrating how important making the distinction between understanding and operation is for business modelling.

2 Why we need business objects' revisionary approach

If business objects change the way we see things, then it seems reasonable to ask why we should go through with this change. Because it involves a substantial upheaval, there should be a pretty good reason to justify it.

2.1 Why entities and attributes are problematic

To appreciate the fundamental reason for a revisionary approach to business objects, we need to understand why entities and attributes are not suited to computer technology. This involves understanding why they developed in the first place.

The entity way of seeing the world (the entity paradigm) is practically prehistoric. It is based on the substance paradigm. This was first formalised by the Ancient Greek Aristotle in the 4th century BC (in other words, over two thousand years ago). We will look in more detail at these two paradigms in *OP1—Entity Ontology Paradigm* and *OP2—Substance Ontology Paradigm*. What is of interest to us here is how pen and paper technology influenced the entity paradigm's development.

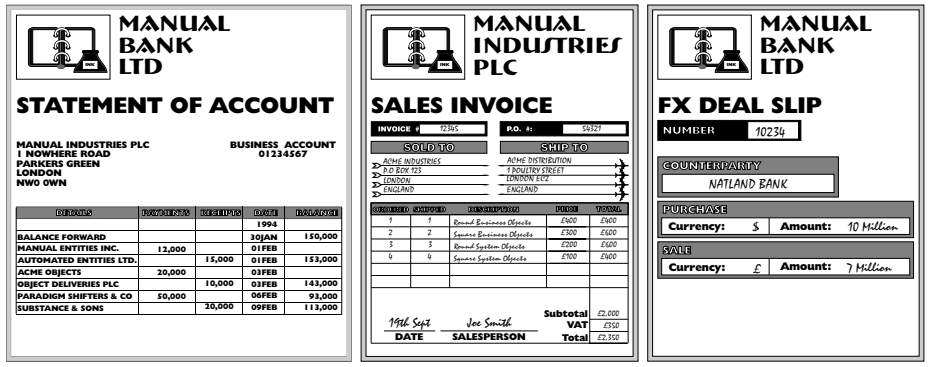
The entity way of seeing things is designed to make it easier to store information, using pen and paper technology. Information about a world seen as entities and attributes is much easier to divide into rows and columns. This division makes it much easier to store on two-dimensional paper. (The ease of use more than offsets the distortions that arise from imposing an entity-view—as we shall see in *OP1—Entity Ontology Paradigm*.)

By contrast, computer technology is not constrained in the same way as two-dimensional paper. It is possible to store computer information in many more ways than just rows and columns. So, when using computer technology, imposing a view based on entities and attributes creates unnecessary constraints.

In these circumstances, we might expect computer information to have thrown off these constraints. However, a moment's consideration reveals that most computer information is still steeped in an entity-view based on paper and ink technology. This is not really surprising because we inherit most of our ways of thinking about information from an age dominated by this old technology.

That is why most computer information fits neatly onto paper forms, such as statements of account, sales invoices and deal slips; examples of which are shown in *Figure AS2-2*. The paper-bound entity way these forms handle information has been imported wholesale into our computer systems.

Figure AS2-2 Forms—products of paper and ink technology



System builders recognise that it is a mistake to use the old manual paper-bound way of handling things when automating a process. Although they recognise this, it is ironic that they are still enchained to a paper-bound entity way of viewing the business.

2.2 Computing technology bringing radical changes

The technology leap from paper and ink to computers is enormous. Yet, the underlying entity paradigm with its rows and columns structure has not yet really changed. As we have just seen, forms such as sales invoices and deal slips have not been transformed into something radically different; i.e., something that looks as if it were based on computing—not paper and ink—technology.

Changes have occurred, but if we look closely, little change has taken place in the basic information structure. The big change is in the efficiency with which they are processed. Automated computer systems process more deal slips (and more sales invoices) faster and more accurately than the old manual paper systems.

Although this is a welcome improvement, i t is still disappointing that we have not yet had the kind of radical change (and the benefits it would bring) one might expect from computer technology. Business objects are now bringing this radical change—some results are described in *MW—The BORO Methodology: Worked Examples*.

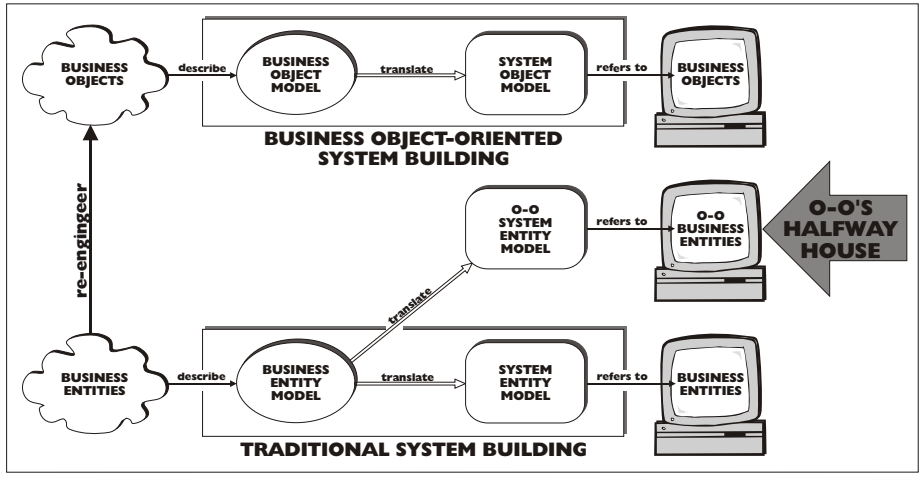
2.3 O-O programming's halfway house

Most O-O system developments are being carried out using an entity view of the business. Without an understanding of business objects, the developers have no choice. This situation is reflected in the many O-O textbooks that suggest using entity modelling for the early stages of system development.

This explains, to some extent, why O-O is currently having more success with objects whose task is to make systems work rather than reflect reality—such as the objects in screen interfaces. For example, O-O has been used to develop impressive graphical user interfaces (GUIs). However these systems have less impressive, more traditional, innards.

Figure AS2-3 explains why. Their innards contain objects that reflect business entities rather than business objects. The result is a kind of halfway house—built from O-O business entities rather than business objects. The figure also illustrates how we need to construct business objects—by re-engineering the business entities.

Figure AS2-3
O-O's halfway house



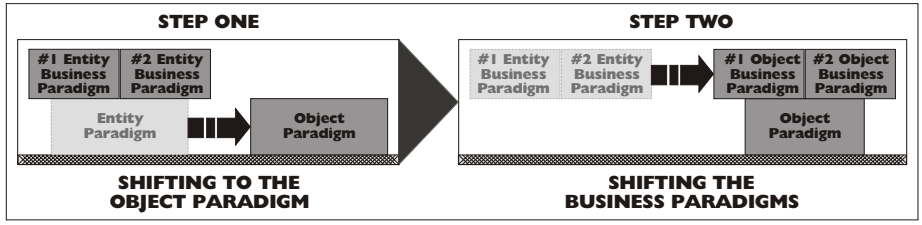
3 What do we re-engineer—paradigms

Figure AS2-3 implies that we need to re-engineer the business entities into business objects. However, this is only part of what must happen. The way we see the business is supported by a whole framework of ideas—a paradigm. It is this that we actually re-engineer.

For our purposes, it makes sense to divide this framework into two levels; a business paradigm and an information paradigm level. The specific objects we are interested in at the business paradigm level will vary from business to business. For example, the securities industry has one group of business level things, such as securities trades; while the oil industry has different business level things, such as oil barrels. Things at the information paradigm level are more fundamental and do not vary from business to business. For example, within the entity paradigm, both securities trades and oil barrels are business entities. This division into two levels makes the re-engineering more straightforward. It falls neatly into two corresponding steps as shown in *Figure AS2-4*.

First, we re-engineer the entity paradigm into the object paradigm (*O—ONTOLOGY Papers* describes how this is done). It is worth noting that this first step does not, by itself, change the way we see the business. At the end of step one (shown in *Figure AS2–4*), the entity business paradigms no longer have a foundation. So, in step two, we re-engineer them into object-oriented business paradigms. These are built upon the new object paradigm that we re-engineered in step one. It is at this stage that we start seeing the business in a radically different way. *MW—The BORO Methodology: Worked Examples* shows how this is done.

Figure AS2–4
Two re-engineering steps



The advantage of this division into two levels is that it separates the information foundations, which only need to be re-engineered once, from the business paradigms, which need to be re-engineered for each business (because they vary from business to business). This means we can re-engineer the information foundations once and for all - as described in *O—ONTOLOGY Papers*. With the new foundations in place, you only need to consider their business paradigm levels when you start to re-engineer your existing systems.

4 What are thebenefits of re-engineering business paradigms?

Once you understand what business objects are, it is relatively easy to re-engineer the business paradigm level of existing systems. But what are the benefits of doing this? When people start re-engineering their business paradigms, the superiority of the object foundations soon becomes apparent. The final business model is not only functionally richer than the original system, but it is significantly simpler and more compact.

4.1 Greater explicitness, increased accuracy andmore re-usable

In the BORO approach, when you start to re-engineer business paradigms, you find that you are constructing a model whose business patterns are both more explicit and more accurate. One important result of capturing more of a pattern explicitly and capturing it more accurately is that it becomes more re-usable. This turns out to be part of a general trend towards greater accuracy in most engineering disciplines. *AS3—What and how we re-engineer* looks at how similar increases in accuracy in manufacturing engineering enabled the development of interchangeable (in other words, re-usable) parts.

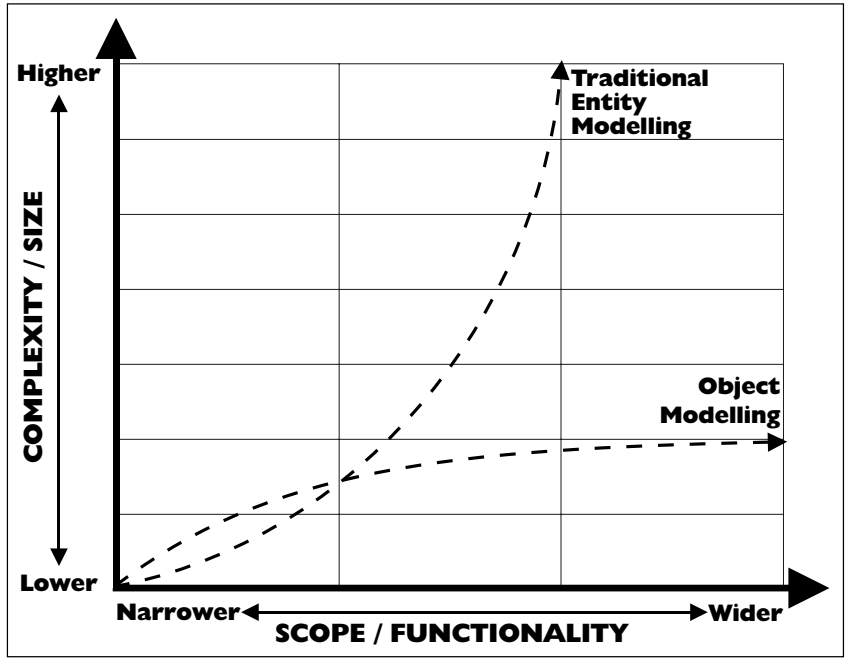
4.2 A substantially more compact business model

As you continue to re-engineer, you begin to realise that not only does the object paradigm enable substantially more compact models, but the larger the scope of the re-engineering the greater the compacting. This is very different from traditional entity modelling (and computer system building). There, when the scope is increased, the overall complexity of the system increases. Each new pattern has to be harmonised with the existing patterns. Each time the scope increases, the task of harmonisation gets more onerous. The traditional rule of thumb is that the more patterns the model contains, the greater the cost of harmonising each new pattern (because there are more patterns to harmonise with).

Business objects handle increases in scope in a very different way. Each new pattern, instead of adding to complexity, provides an opportunity for compacting a number of patterns into a single, more general, pattern and so creating a simpler model. Adding additional new patterns creates further opportunities to compact, generalise and simplify the model.

The more effective way in which the object paradigm deals with increases in scope is illustrated in *Figure AS2-5*. *MW—The BORO Methodology: Worked Examples* provides examples.

Figure AS2–5
Increases in scope



5 Summary

To briefly summarise what has been said in this paper:

- Underlying the BORO approach is a core issue—how can we accurately reflect the business in a model and so in a computer system? *The BORO Working Papers* explain how business objects can tackle and resolve this issue by changing the way we see things.
- People currently see things in terms of entities and attributes. These were developed for use with paper and ink technology and are unsuited for computing technology. Business objects, by contrast, can take full advantage of computing technology's potential.

We need to re-engineer our business entities into objects. This is done in two steps; a re-engineering of the current entity information foundations to object foundations and then a re-engineering of the business paradigms from entity to

object foundations. In many ways the task of *The BORO Working Papers* is to help people make this first step and prepare them for the second step - re-engineering thier existing systems.

The re-engineering brings enormous benefits. It enables better business models—and so computer systems—to be built. These are simpler, more compact, more explicit, more accurate and more re-usable.

Before they can build these models, people need to understand what business objects are and learn how to apply them. *MW—The BORO Methodology: Worked Examples* helps you acquire the skills in applying business objects by taking you through a number of worked examples.

BORO Working Papers - Bibliography

The BORO Working Papers

Volume A
A—The BORO Approach

Book AS
AS—The BORO Approach: Strategy

AS1—*An Overview of the Strategy*
AS2—*Using Objects to Reflect the Business Accurately*
AS3—*What and how we re-engineer*
AS4—*Focusing on the Things in the Business*

Volume - O
O—ONTOLOGY Papers

Book - OP
OP—Ontology: Paradigms

OP1—*Entity Ontology Paradigm*
OP2—*Substance Ontology Paradigm*
OP3—*Logical Ontology Paradigm*
OP4—*Business Object Ontology Paradigm*

Volume - B
B—Business Ontology

Book - BO
BO—Business Ontology: Overview

BO1—*Business Ontology - Some Core Concepts*

Book - BG
BG—Business Ontology: Graphical Notation
Constructing Signs for Business Objects

Graphical Notation I
BG1— *Constructing Signs for Business Objects*
Graphical Notation II
BG2— *Constructing Signs for Business Objects' Patterns*

Volume - M
M—The BORO Re-Engineering Methodology

Book - MO
MO—The BORO Re-Engineering Methodology: Overview

MO1—*The BORO Approach to Re-Engineering Ontologies*

Book - MW
MW—The BORO Methodology: Worked Examples

Worked Example 1
MW1—*Re-Engineering Country*
Worked Example 2
MW2—*Re-Engineering Region*
Worked Example 3
MW3— *Re-Engineering Bank Address*
Worked Example 4
MW4—*Re-Engineering Time*

Book - MA
MA—The BORO Re-Engineering Methodology: Applications

MA1—*Starting a Re-Engineering Project*
MA2—*Using Business Objects to Re-engineer the Business*

Book - MC
MC—The BORO Re-Engineering Methodology: Case Histories

Case History 1
MC1—*What is Pump Facility PF101?*

THE BORO APPROACH: STRATEGY - 2

USING OBJECTS TO REFLECT THE BUSINESS ACCURATELY

A-O

A

B

C

D

E

I

J

O

P

R

U

Y