

8-15-1997

# OOT: The Tortuous Path to a Bright Future

Jean\_Pierre Kuilboer

*University of Massachusetts*, [Kuilboer@umbsky.cc.umb.edu](mailto:Kuilboer@umbsky.cc.umb.edu)

Follow this and additional works at: <http://aisel.aisnet.org/amcis1997>

---

## Recommended Citation

Kuilboer, Jean\_Pierre, "OOT: The Tortuous Path to a Bright Future" (1997). *AMCIS 1997 Proceedings*. 205.  
<http://aisel.aisnet.org/amcis1997/205>

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 1997 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact [elibrary@aisnet.org](mailto:elibrary@aisnet.org).

# **OOT: The Tortuous Path to a Bright Future**

[Jean pierre Kuilboer](mailto:Kuilboer@umbsky.cc.umb.edu)

Kuilboer@umbsky.cc.umb.edu

100 Morrissey blvd, Boston, MA 02125

University of Massachusetts, Management Science and Information Systems

## **Abstract**

The gap between the recognition of the importance of Object-Oriented Technologies (OOT) and their wide adoption is still wide. While the practitioner's press has endorsed the common view that the opportunities are ripe for firms adopting the object paradigm, both university higher education systems and the industrial sectors have been slow to fully endorse the shift. The following paper will outline some of the most obvious reasons for the slow transition, locate some of the potential venues for remedy, and highlight a possible path for adoption.

The commonly mentioned object-oriented techniques are composed of an eclectic set of techniques, methods, and tools. The short paper is intended to outline their convergence toward a wide transition to objects.

## **Object Oriented Database (OODBMS)**

In the late 1980's, I visited a few of then promising object-oriented database software developers. Firms such as Ontos, Object Design, and Gemstone were likely to be future database market leaders. The ORION project started in 1985 (Kim, 1990) at MCC (Microelectronics and Computer Technology Corporation) has triggered an interest for Advanced Computer Technologies and the object oriented database approaches. Amid speculation of the time, the market reality has contradicted these predictions. By 1997, the relational model is healthy and vendors such as Oracle, IBM, Sybase, and Informix are still dominating the database market by a factor of 10. Joint-venture capitalists have in the meantime been quite disappointed and moved onto newer pastures (i.e., the Internet ventures or bio technology). However, today there seems to be a renewal of interest for OODBMS and one wonder if 1997 will finally be the OODBMS year. Also, while the purist declaim the facts, it could well be that the innovation will come from the traditional providers; Oracle will release Oracle 8.0 with some Object-Relational features and support for spatial, text, graphics, video data. The Oracle approach implements object-oriented polymorphism and inheritance. Informix following its merger with Illustra has released its new Universal Server including the features such as inheritance and polymorphism and support for complex types through third party DataBlades (external program allowing interpretation of stored buckets of bits). Similarly IBM has the intention of adding object-relational features in forthcoming DB2 Universal Database Version 5. As the Informix approach they will offer external extender allowing interpretation of BLOB and support the clustering, polymorphism, and inheritance aspects of the object oriented paradigm. Adoption of some of the OO features is therefore almost inevitable in the present marketplace. Pure object-oriented database systems have also reached better recognition as potential solution for mission critical system. Reports start to appear in the practitioner press (e.g. Distributed Object Computing Magazine)

that OODBMS are the foundation of systems competing with traditional transaction processing systems (e.g. TPF). Air France is using globally a system based on Versant Technology and developed by SABRE the AMR subsidiary.

### **Object Analysis and Design**

The deployment and adoption of a new paradigm is a lengthy process. In information system the result will not be a full transition but the addition of a new layer of technology. The epoch of the structured techniques preceding the current enthusiasm for anything Object-Oriented had a bumpy evolution. Even though, the programming languages were more stable, since the COBOL language dominated Business programming for the past 25 years, it took a long time for the structured techniques to be widely used.

With the COBOL legacy, the Structure techniques had to progress only on one front, i.e., from the Implementation back toward requirement analysis. The structured techniques with their associated notation based on the data flow diagram (DFD), Structured Charts (SC), and later the entity relationship diagram (ERD) associated with the increasing consideration for the data aspects were fairly straightforward. The difficulty resided in scaling up from simple system to highly complex development. From a teaching point of view, support from book publishers increased through the 1980 culminating with well-documented case studies, exercise sets, and supporting CASE tools. If one chose to ignore the Information Engineering movement, the field changed at a pace that could be accommodated by both teachers and the Information system function in business.

In the early 1990's , the background of the object-oriented authors reflected their origin and bias (either CS or Engineering). Teaching OT from a business perspective through the Rumbaugh et al (1991), Booch (1991) and Jacobson (1992) books, the MIS faculty had to adapt to approaches replete with examples such as ADA traffic management system, Object-Pascal Geometrical Optics construction kit, or electrical machine projects. While the financial institutions and wall street were the early adopter of the new technology, most examples were extracted from other disciplines (i.e electrical engineering). The exceptions were books targeted at managers such as Taylor (1990), offering a clear overview but no in-depth method coverage. The present stance has slightly changed. Books are appearing, targeted at the MIS side of the business, and the adoption of more OOSD should ensue. Examples, aimed at the higher education market would be "Systems Analysis and Design and the Transition to Objects" by Donaldson-Dewitz (1996) or "Object-Oriented Analysis: Objects in Plain English" by D. Brown. When the level of support for OT within the new CASE tools is sufficiently high, the conversion to the OOSD will slowly occur; it will be accelerated by synergy with object-oriented programming if the deployment of Java gains some strength.

Adoption of OOSD by the educational system and industry will accelerate with the consolidation occurring on the methodology side. Two proposals have been made to the OMG technical committee: the first is offered by the supporters of UML (Unified Modeling Language) the method produced by the Booch/Rumbaugh/Jacobson trio, it

represents a synthesis of the three approaches (OMT, BOOCH, Objectory/Use cases) into one method. The UML proposal is supported by industry giants such as Hewlett-Packard, MCI, Microsoft, Texas Instrument, Oracle, and Unisys. The competing method, Open Modeling Language ( OML) advocated by Platinum technology has slightly less support but is not conceptually very different from the UML offering.

The main difference lies in the vested research made by the respective supporters. The convergence will increase the chances for second sourcing of a single method (e.g. Microsoft and Rational) and provide incentives to write relevant books, thereby leading academic support. Publication of relevant supporting material has been somewhat delayed by the reality of lucrative tour by the leading methodology "gurus" but first wave will be forthcoming shortly.

Time has definitively come to stop publishing an assortment of proprietary methodological exercises which have not pragmatic chances to ever be adopted in practices and do not offer any clear benefits over proposed methods.

### **Programming Language**

The burgeoning of Object oriented techniques occurred first with the increasing adoption of 'C' followed by its successor 'C++'. Historical perspective on initial OO languages such as SIMULA is relevant in pinpointing the lengthy timeline of the adoption curve. The evangelization accomplished by the advocates of Smalltalk and XeroxParc researchers was also the starter of the current revolution.

The wide adoption of C++ in a number of industries and its standardization have influenced the leading authors in rewriting their books using the language. Probably more is to come with the highly publicized interest and the latest hype about JAVA a close relative to C++. Remarks similar to the OOSD will easily apply to the OO programming discipline. Numerous books have been written for C++, most of them targeted at the self-learner or specific compilers (e.g., Illustrated by title series such as "Learn C++ in 7..21 days") or the computer science curriculum. While enrollments in MIS were rising and industry interest for C++ is at its highest, very few books have been targeted at the business students or the business user. Given the image of "C" as a system programming tool, the C++ position has been slightly hindered on the business side. The slow adoption is also explained by the difficulty in either teaching or learning the language without a solid background. The emergence of a few books (e.g., "Building Business Applications Using C++" by L. Garnett or "C++ and the Object-Oriented Paradigm" by J.L. Harrington) aimed at the business side has only slightly improved the situation but support for business examples remains dim. C++ is still a difficult language to learn and is unlikely to succeed as an initial language. Up to the ramp with C++ the JAVA language could be the likely savior but most organization and schools have not yet converted to the supported platform (i.e., UNIX, Windows95/NT, OS2) and the steep hardware requirements and few instructor are conversant with the language.

### **Organizational Aspects**

The Management Information system discipline was associated with a strong growth in information systems development in the 1980's. The convergence of a number of factors supported its growth at the time. A common set of structured techniques was adopted (i.e., Yourdon & De Marco, or Gane & Sarson). In short order other facilitating factors were:

The recognition by the AACSB that computer literacy was essential to a well-rounded business education. The immediate response to this trend was the rapid rise in enrollment in IS programs. The conversion of a number of dynamic existing educators to the IS discipline through AACSB-sponsored Summer camps.

A prosperous economic period was able to absorb the newly trained IS workforce.

The computer programming field was fairly stable. While the 'C' programming was making some inroad in the research laboratories and engineering, ADA was making strides for DOD contracts, COBOL was definitely dominating the industrial/business space.

Computer science programs were likely flourishing due to the very promising job prospects for the graduates.

The late 1980's and early 1990's illustrated a completely different picture and a negative outcome for IT. A few examples are easy to come by:

While prospects in IT were still numerous the general press focused on highly publicized failure and reversal such as large layoffs at both IBM and Digital Equipment, the bankruptcy of Wang Computers.

The attractive positions in higher education both in research and teaching were starting to seriously dry out. From a high ratio of positions offered per candidate in the mid 1980's to a bleak situation in the early 1990's, the reversal was definitively a boom to bust case in students' enrollment.

At the undergraduate level the computer science students were increasingly reluctant to face the burden imposed by an increasingly competitive and decreasingly rewarding market. Gone were the easy offers for stock option in dynamic startup, the time had come to send one hundred resumes without positive response.

The model curriculum set by ACM, AIS, and DPMA only minimally mentioned object technology.

The downsizing movement, a decrease in enrollment and the AACSB deemphasis of computer education had limited MIS/CIS to constrained department growth.

When the introduction of OODS occurred. The dynamic faculty from the 1980's had got tenure and had lesser incentive for another reconversion.

## **Reasons for Hope in Industry and Academia**

Presently, looking for qualified faculty and relevant teaching materials is still challenging. While basic books could be easily accommodated there was and still is a dearth of teaching and case books. However, publication groups have appeared specializing on the topic (i.e. SIGS). End to end teaching OOT business case remain scarce while there is a plethora of structured techniques' offerings (all disincentives to change). On the bright side many educators are involved in consulting and are well aware of the incentive offered to OOSD adopters.

### **Conclusion**

The industrial and service sectors, while increasingly aware of some significant benefits in the domain of maintenance and flexibility, still lack confirmatory solid evidence of the advantages provided by the OO techniques and thus remain in a pilot mode. Some sectors have shown some promises, using OO as competitive advantage due to the flexibility of change provided by this approach. Endorsement by firms such as IBM, SUN, and Microsoft add some much needed substance to the claims.

The present paper parallels the reality of the quickly changing marketplace with a high complexity not well understood even by its main actors (i.e. Microsoft, IBM, SUN). While detractors claim that OO is offering nothing new, the mere quantity of different approaches in the interdependent segments (Analysis, Design, Programming, testing, measurement, maintenance, database) is rendering a very complex picture. Potential adopter should be well aware of the interdependencies. Introduction of either OOA/OOD or Object Programming in the MIS curriculum will have effects which should not be understated. Because of high synergy between the segments (ie. OOA/OOD, Programming, OODBMS) it is less likely that isolated adoption will raise high benefits. On the other hand an overall adoption could imply high risks. While promises are certainly worth the effort, exploration is recommended first before jumping otherwise OOT could become the next unfulfilled hype.

Once the object approach reaches a more prominent status in industry the self propelling movement will accelerate academic acceptance. Leading IS curriculum, while still lagging the industry, will soon reflect the changes to the new paradigm.

### **References**

Booch, G. *Object-Oriented Design with Application*, Benjamin-Cummings, Redwood City, CA, 1991.

Brown, D. *Object-Oriented Analysis Objects in Plain English*, John Wiley & Sons, Inc., New-York, NY, 1997.

Donaldson-Dewitz, S. *Systems Analysis and Design and the Transition to Objects*, McGrawHill, New-York, NY, 1996.

Garnett, L. *Building Business Applications Using C++*, Addison-Wesley, Reading, MA, 1997.

Harrington, J. *C++ and the Object-Oriented Paradigm, an IS Perspective*, John Wiley & Sons, Inc., New-York, NY, 1995.

Kim, Won, *Introduction to Object-Oriented Databases*, MIT Press, Cambridge, MA, 1990.

Rumbaugh, J. Blaha, M. Premerlani, W. Eddy, F. And Lorensen W. *Object-oriented Modeling and Design*, Prentice Hall, Englewood Cliffs, NJ, 1991.

Taylor, D.A. *Object-Oriented Technology a manager Guide*, Addison-Wesley, Reading, MA, 1990.