Despite its decade long history, object database technology has never entered mainstream system development. In this work, I look at the background of the slow adoption, and the possible future outlook of this technology based on my personal experience and the published literature. Object databases represent a revolutionary new technology and provide superior storage facility for complex data structures and types. They also enable close language binding and a unified development process. It is a mature technology with advanced database management and development features and has several proven and robust deployment examples. Besides its current technical excellence, this technology is also demonstrating future potential through such emerging technologies as Java, Application servers, and XML. The past failure of object databases to proliferate the market was mainly due to unawareness, lack of skills, and the overwhelming existing investment in relational systems. As these factors are changing by the end of this century and new technology adoption is accelerating, object databases are looking forward to a slow but sure take off.

Technical and Market Viability of Object Database Technology

It might be a somewhat surprising fact that object databases have been around for as long as a decade. Since their appearance, market researchers and technology experts have been making confident projections that they would conquer the landscape of software industry. Indeed, information technology has gone through an amazing series of changes, but object database technology has yet to enter
mainstream computing. Is this due to the overall weakness of this architecture, or is it rather the result of slow adoption of new technology?

By the second half of the 1990s, more and more new software projects selected object-oriented (OO) technology as the basis for their development. This paradigm shift naturally drew increased deployment of OO analysis, design, and language tools. Objects became the basic unit of processing, posing new demands on data storage requirements. The increased complexity of data structures and types also affected the architecture of modern databases. Aggressive changes in the business environment, such as globalization and deregulation, are demanding greater flexibility and complexity of supporting information systems. As a result, their underlying data pool also needs to reflect these architectural changes.

Once again, why has object database technology not entered mainstream enterprise development despite the favorable technical environment? Do object-oriented database management systems (OODBMSs) offer a different and improved technology and architecture, or are vendors only trying to take advantage of the increasing popularity of the OO paradigm? Is this technology mature and scalable enough for prime time? Will future technologies leverage object databases fostering their widespread application, or will they suffice with the relational model? These are the questions that I will attempt to address in this chapter. This review is based on both personal experience in OO development and the extensive body of OO literature.

**Technical and Architectural Validity**

The operational principles of today’s OODBMSs were inherited from their early ancestors. Object databases were developed to handle complex, interrelated data structures. They were first used in computer-aided design (CAD) applications where relational databases could not provide the performance needed. Object databases provided persistent storage in a single user environment with high performance. These early systems could store objects, classes, associations, and methods. In the late 1980s, commercial vendors started developing independent OODBMS products. They added database capabilities to support multi-user, distributed applications. In the second half of the 1990s, the increasing popularity of OO languages has renewed the attention toward object databases. Next, I will look at the core technology of object databases, especially those basic architectural features that validate and differentiate OODBMSs. These technical improvements over relational technology will help to understand the very need for this technology in certain areas of information technology (IT).

**Complex Data Relationships**

The efficient handling of complex data relationships is one of the greatest advantages of object databases over their relational counterparts. This improvement was an important driving force in the adoption of OODBMSs for early users. Information access patterns differ from application to application. Standard business programs select a small subset out of a large amount of data for processing, just like a typical payroll system. Whereas, in other applications, such as in engineering programs, a larger amount of data is being manipulated by constantly navigating
Related Content

Using Linguistic Patterns to Model Interactions
[www.igi-global.com/chapter/using-linguistic-patterns-model-interactions/27358?camid=4v1a](http://www.igi-global.com/chapter/using-linguistic-patterns-model-interactions/27358?camid=4v1a)

Software Reuse and Object Technology
Jane Fedorowicz and Denis Lee (2000). *Object Oriented Technologies: Opportunities and Challenges* (pp. 77-88).
[www.igi-global.com/chapter/software-reuse-object-technology/27348?camid=4v1a](http://www.igi-global.com/chapter/software-reuse-object-technology/27348?camid=4v1a)

Agile Project Controlling and Effort Estimation
[www.igi-global.com/chapter/agile-project-controlling-effort-estimation/25642?camid=4v1a](http://www.igi-global.com/chapter/agile-project-controlling-effort-estimation/25642?camid=4v1a)
Introduction to the Objects Approach in Software
www.igi-global.com/chapter/introduction-objects-approach-software/18731?camid=4v1a