

The Component Software Solution: OLE's COM

The Component Object Model provides a means to address problems of application complexity and evolution of functionality over time. It is a widely available, powerful mechanism for customers to adopt and adapt to a new style multi-vendor distributed computing, while minimizing new software investment.. COM is an open standard, fully and completely publicly documented from the lowest levels of its protocols to the highest. As a robust, efficient and workable component architecture it has been proven in the marketplace as the foundation of diverse and several application areas including compound documents, programming widgets, 3D engineering graphics, stock market data transfer, high performance transaction processing, and so on.

The Component Object Model is an object-based programming model designed to promote software interoperability; that is, to allow two or more applications or "components" to easily cooperate with one another, even if they were written by different vendors at different times, in different programming languages, or if they are running on different machines running different operating systems. To support its interoperability features, COM defines and implements mechanisms that allow applications to connect to each other as *software objects*. A software object is a collection of related function (or intelligence) and the function's (or intelligence's) associated state.

In other words, COM, like a traditional system service API, provides the operations through which a client of some service can connect to multiple providers of that service in a polymorphic fashion. But once a connection is established, *COM drops out of the picture*. COM serves to connect a client and an object, but once that connection is established, the client and object communicate directly without having to suffer overhead of being forced through a central piece of API code as illustrated in Figure .