

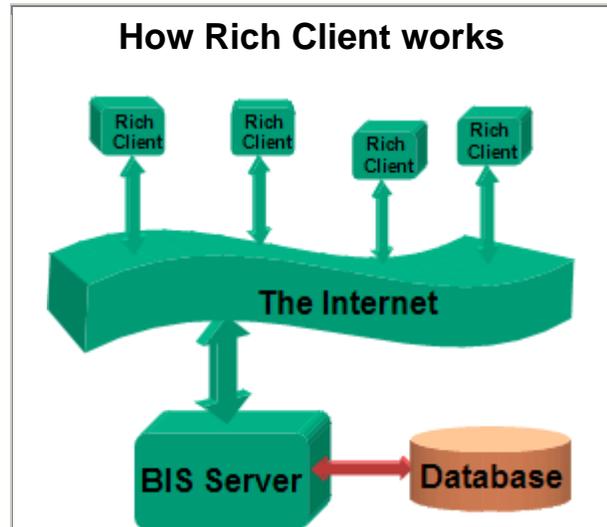
## Base One's Rich Client Architecture

Base One provides a unique approach for developing Internet-enabled applications, combining both efficiency and ease of programming through its "Rich Client" architecture. With this architecture, it is possible to build graphically intensive programs that look and perform like sophisticated client/server Windows applications, but where data is actually stored and retrieved across the Internet. At the same time, Base One makes it easy to develop such applications on a laptop or desktop and deploy them over the Internet without any changes.

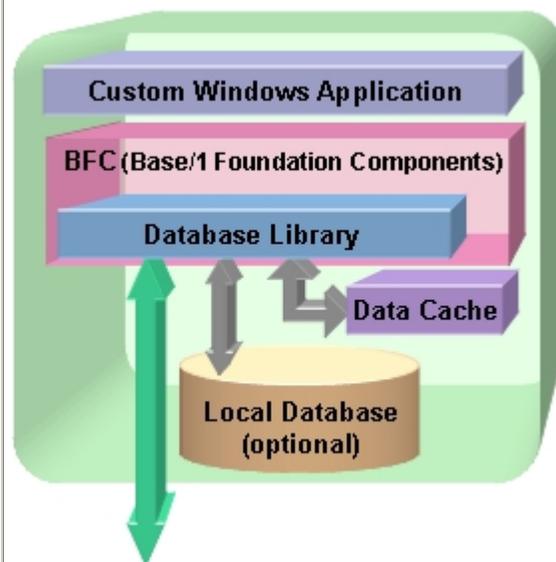
In contrast to thin client models, a rich client application shifts the burden away from shared network resources and server machines onto client computers, for better overall performance and application flexibility. While many applications are adequately handled by thin client architectures, some applications are much better suited to Base One's Rich Client model. (See table below for the types of applications that benefit the most.)

### A Service Oriented Architecture

Like systems based on Web Services, Base One's Rich Client model is also a type of Service Oriented Architecture (SOA). It comprises independent, co-operating components (services) that can be distributed within or outside of an organization's physical boundaries and security domains, using varying platforms and programming languages.



In Base One's model, each Rich Client looks like a normal database application - because a Base One Internet Server (BIS) acts as a transparent proxy for a database across the Internet.



A Rich Client consists of a custom application written in Visual C++, C#, VB or VB.NET, using Base One's class libraries (BFC), and running on any modern version of Windows (2000, XP, 2003, or Vista).

However, unlike systems that rely exclusively on Web Services, SOAP, and other XML messaging technologies, Base One models the interactions between Rich Clients as pure database operations that function against local or remote databases. Instead of requiring explicit, application-level calls to remote services, the Base One Database Library automatically handles network communications, transparently.

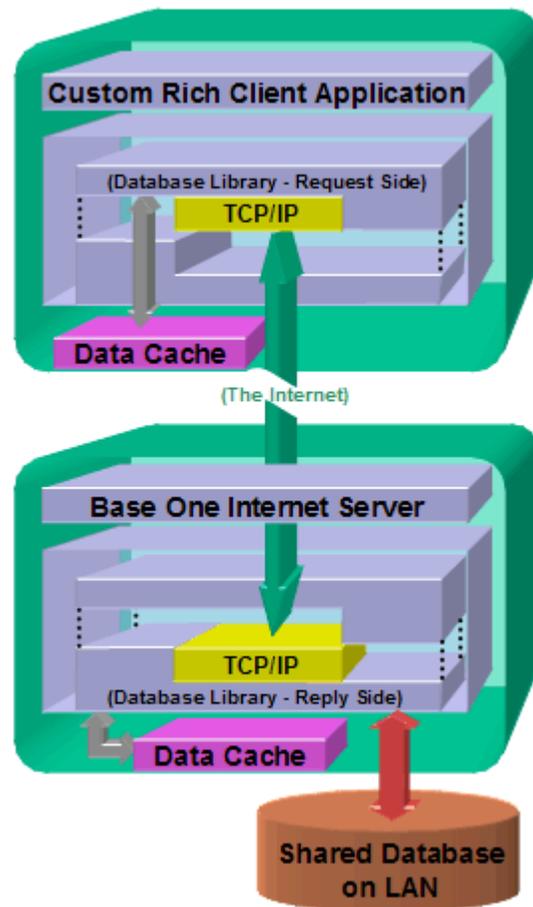
### Database-centric Design

Base One's Rich Client architecture is built around a high-level abstraction of well established database concepts, familiar to anyone who has developed traditional SQL client/server applications. The Rich Client model takes those familiar concepts further, by wrapping the low-level details into a platform-neutral layer of database functionality with an active Data Dictionary, which can be used from a variety of programming languages, with different back-end DBMS products, locally, on a LAN, or across the Internet.

This versatility is made possible by Base One's patented design, which incorporates a core of TCP/IP communications functions into the heart of the Database Library. With the Base One Internet Server (BIS) acting as a proxy for remote database access, applications need not know where the data is coming from. Application programmers simply work with Base One Record Sets, which can be manipulated conveniently in either a compact, binary representation, or in XML format.

### How Rich Client works (continued)

The Database Library employs a high-speed memory cache, and it can further accelerate access by buffering operations against a local database. Peer-to-peer communication with other Rich Clients is achieved through data storage and retrieval functions, so any such communications are entirely transparent to custom application programs.



Extending the model to a third tier, the core TCP/IP communications functions built into the Database Library support database access across the Internet, without any alteration of custom Rich Client applications.

As a security feature, Base One's design supports custom encryption and decryption of all Internet traffic that passes through the Database Library's TCP/IP core. (Examples of custom encryption are included in the sample projects that come with Base One's programming tools.)

### **Performance Advantages**

When dealing with large volumes of data, Base One's Rich Client architecture is substantially more efficient than XML-based systems, for several reasons. First of all, Base One's standard representation of Record Sets is considerably more compact and far simpler than XML. Record Sets require less processing, since there is no need for complex recursion and extra encoding / decoding steps between the application program and the database.

Another advantage of Base One's design is its ability to make better use of local temporary storage and automatic caching, thereby allowing more work to be performed, much faster, on client machines. For example, by caching Data Dictionary metadata, Base One avoids including field information in data transfers and redundant requests to database servers.

### **Programming Advantages**

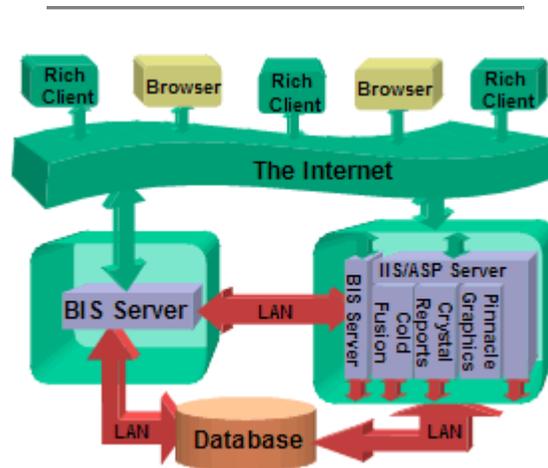
What is remarkable about Base One's approach is that it's not only more efficient, but also much easier to program.

### **How Rich Client works (continued)**

Base One's peer-to-peer TCP/IP functions work through the built-in communications facilities of Windows itself, so there is no requirement for a conventional web server (such as IIS) on either the client or the server. A standard TCP/IP connection for Internet or intranet communication is all an application needs to send database requests and receive responses across the network to or from a Base One Internet Server (BIS).

Performing database access through BIS eliminates the need for the database's client software to be installed on client workstations or application servers. By using the Base One Database Library, no vendor-specific database client software has to be installed on that computer - only regular TCP/IP messaging capabilities. This allows applications to do efficient, secure database access with no direct database connectivity, and therefore no need for transmission of database passwords.

On the server side, no custom programming is required, because BIS (itself a form of Rich Client application, provided by Base One) can do its job efficiently, without knowledge of custom application details.



In addition to supporting Rich Client applications,

There is no awkward paradigm shift. The Base One's Database Library is a logical extension of the familiar relational database model, but with a high level of abstraction that frees programmers from concern about numerous low-level details. In addition, the complication of multi-tier, server-side programming can be entirely dispensed with, because a Rich Client application requires no application-specific programming on the server side.

### **How Rich Client works (continued)**

Base One's Internet Servers can work with conventional Internet servers, using ASP or ASP.NET to support a mix of both thin clients and rich clients. Further easing the deployment of Rich Client applications, Base One does not require installation of costly DBMS software, reporting, and graphics packages on each client computer. These services need only reside on a relatively small number of BIS servers.

Since a single design can be used in altogether different environments with minimal reprogramming, Base One applications are easier to maintain and adapt. The active Data Dictionary also gives programmers the flexibility to adjust for changing system requirements and database structure as conveniently as possible. Other Rich Client components, including a Command Processor, diagnostic tracing, logging and error handling facilities, and mechanisms for automated transaction processing and failure recovery further contribute to the ease of implementing, debugging, and administering applications.

---

## **Applications that can benefit from Base One's Rich Client Architecture**

| <b>Application Type</b>                      | <b>Benefits</b>   |
|--|---|
| <b>Data-intensive graphical applications</b> | Rich Client applications may be highly refined, graphical applications that perform well, even over relatively slow Internet connections. Free from browser limitations, Base One's local database and caching capabilities allow programs to manage large amounts of information, combining the speed of a local hard disk with the timeliness of a shared database. Thus the quality of user experience in accessing an Internet site through a Rich Client application can be similar to that of using a visually rich, Windows encyclopedia like Microsoft Encarta. |

|   |   |
|---|---|
| <p><b>Applications that support thin client web sites</b></p>   | <p>Web sites that provide access to thin client users (e.g. through browsers or Citrix) can benefit from using Rich Client applications to perform high volume, back-office and administrative tasks efficiently, with minimal disruption of operations. Base One's Batch Processing Facilities allow these background database jobs to be scheduled, monitored and controlled, either locally or remotely.</p>   |
| <p><b>Large-scale distributed processing applications</b></p>   | <p>Rich Clients are the building blocks of Base One's architecture for scalable, high performance grid and cluster computing. By leveraging underutilized computers and fully exploiting commercial DBMS capabilities for efficient, reliable transaction processing and secure data storage, Base One's design yields cost-effective solutions to very large distributed computing tasks. Base One grid and cluster computing solutions are practical and easy to deploy, because they run on any version of Windows (from Windows 98 on up), without extraordinary hardware or software requirements. With a small footprint and the simplest possible XCOPY deployment, it's easy and economical to distribute a Rich Client application across very many computers.</p>   |
| <p><b>Applications that interface with legacy databases</b></p> | <p>In situations requiring the exchange of large volumes of data between loosely-coupled production databases, Rich Client architecture offers an easy, efficient, and secure solution with the least risk of impacting ongoing operations. For systems that are based on J2EE, CORBA, or mainframe technology, one can use either Base One's XML Record Sets (for interoperability) or Base One's compact Record Sets (for speed), or a combination of these techniques. As long as one or more Base One Internet Servers (BIS) can communicate with a database server, no additional server programming is required. Remote Rich Client applications have no need to communicate directly with the database server, because BIS machines do the actual database access and they can be located within a secure data center.</p> |

|  |  |
|--|--|
| <b>Applications that require specialized equipment across the Internet</b> | Rich Client computing applications can connect specialized equipment (like medical devices, industrial process controllers, alarm systems, and even household appliances) to the Internet. These types of devices have little use for browsers and often have programs that allow for direct connection to PCs. By using Base One's Internet Server for database access, Rich Client applications provide an inexpensive, easily deployed alternative to writing complex programs for connecting hardware and communicating across the Internet.   |
| <b>Smart Client applications</b>   | Microsoft's Smart Client model, using the latest Visual Studio.NET framework and developer tools, works even better on data-intensive applications, when coupled with Base One's Rich Client components. By using Base One Record Sets in place of ADO.NET, it is possible to dramatically speed up a Smart Client, as well as simplifying support of either connected or disconnected operation. Base One makes it much easier to construct intelligent hybrid applications, seamlessly merging a large, relatively static body of information with a smaller portion of dynamically updated information picked up across the Internet. Instead of using one interface (ADO) for local data and another (Web Services) for remote, Base One's Database Library handles either case through a single, common interface, and it works across all of the major databases, without custom application server programming. |