Ebook : Overview of application development.

All code from the application series books listed at:

http://www.vkinfotek.com

with permission.

Publishers: VK Publishers
Established: 2001
Type of books:

“Develop your Own Series”

Database Software, Accounting software, Stock Management software

Tools: Microsoft Tools – VS, C#, VB, SQL Server

Develop your own Database Software

In this ebook, I will run you through the important steps involved in developing client/server database application software. Most of the applications we see today, technically speaking, are client/server application softwares.

Before we go further, let us know more about client/server applications.

Client/Server applications have become predominant in enterprise wide computing. This is because Client/Server computing provides an open and flexible environment where mix-and-match is the rule.

A Client/Server network is used to accomplish many business needs. They are:

a. To record accounting transactions like purchases, sales, receipts, payments in a company. Note that, different users enter these transactions simultaneously using the client computer. The client computer may be running a windows application or a web application. The Client/Server application, should support recording of these transactions into a central Database server confirming to all security and database integrity requirements of an accounting application.
b. To extract business intelligence reports from the data in the database server.
c. To enable users to share resources like printers and fax machines.

As enterprise wide computing becomes more sophisticated, the CIO of a company has a challenging job involving maintenance of hardware and networks, upkeep issues like registry clean up, setting up and updating policies, regular backups, etc., ensuring that the Client/Server business application is up and running. A complex job like this requires an understanding and a grasp of all the issues like hardware, networks, network operating systems, protocols and more.

Client / server application development can be broadly categorized as:

Installing and setting up windows operating system for development and production environment
Installing and setting up database server - sql server
Front end design and development
Back end database design and development

Understanding the Client/Server architecture

On the client side of this architecture, we find a fairly simple Front-end application. On the server side, we find an intelligent engine. The server is designed to accept queries from the front-end application – usually in the form of calls to stored procedures and return the requested information. We will see now in detail what are the responsibilities of client and a server.

Client Responsibilities

The client application is responsible for connection management, data capture, data retrieval, data presentation, and error management.

Connection Management: Client/Server applications establish a connection with the server to submit queries, retrieve result sets. In this architecture, the client makes a hard connection with the server over a local area network (LAN) or a wide area network (WAN) – even if the connection is

with a SQL Server, running on the same system. The client application's connection is authenticated by the LAN and SQL server with a user-supplied or application-generated login ID and password. In some cases, Microsoft Windows provides a degree of connection security management. That is, using domain-managed security, client applications need not provide additional user names and passwords to SQL Server—once they are authenticated by windows. The client must also deal with connection problems caused by trouble on the server, on the network, in the application itself.

Data Capture: The client presents forms for the user to fill in with data. The client validates data values before they are sent to the database. This pre-validation often involves cross-checking form fields against other fields, either on the form or in the database. Validation can take place as fields are filled in or completed, or as the form is committed. These validation criteria are often referred to as client-side business rules.

Data retrieval: The client submits queries to the database engine for processing and retrieves the result sets, as required. Another responsibility of the client is to manage data returned to the application.

Data Presentation: The client application is responsible for displaying results from queries, as needed. This task might involve filling a Grid or Listbox control.

Error Management: The client is also responsible for trapping and dealing with the errors. Effective and comprehensive error management is the sign of a successful client application.

The Server's responsibilities

In any client/server implementation, the server is not just a data dumping ground. The server is also responsible for intelligent resource management, security management, data management, query management and database system management.

Resource management: The server is responsible for managing its own resources. These resources include RAM, connections, disk space, CPU time, threads, and a set of caches or queues. If the server has to compete for resources with other Windows 2003/ NT services, its job is made that much harder. For example, if the server must also act as a print server or domain controller, the SQL Server operations will become slow.

Security management: The server prevents unauthorized access to itself and the database while permitting guarded access to those with valid permission.

Data management: The server is also responsible for the validity and integrity of the data sent to the database system from the client application.

Query management: The server processes SQL queries from the clients, which involves syntax and object checks and compilation of a query into a valid and efficient processing plan.

Database system management: In managing the database system, the server manages all connections to the database. The server also maintains tables, indexes, procedures, rules, triggers, data types, list of valid users, and other database objects.

Let us list and discuss the important topics we should be familiar with concerning the windows operating system. Here my focus will be on discussing topics, which are more relevant and useful for a programmer who is into application software development.

Windows provides two models of network administration. They are workgroup (peer-peer) and domain model (client/server). The domain model, which involves usage of active directory services, is more prevalent and suitable for large networks and where centralized administration is must.

Setup a domain model for a Client / Server Network

A domain model provides centralized administration because user's information is stored centrally. In the workgroup model to access resources on two machines, it is necessary to have two user accounts one on each computer. The domain model provides a single logon process for users to gain access to network resources.

Active Directory Service is the service used to implement the domain model. With the ADS, all the information needed to use and manage printers, shared folders and other resources are stored in a centralized location and the process of locating and managing these resources is simplified. The term directory in ADS refers to a database, which stores information of network objects or resources.

The ADS can be installed on the computer, which is running Server OS windows 2002 or 2003. The process of installing Active Directory on a Windows Server 2003 server computer begins by launching the Active Directory Installation Wizard often referred to by its executable file, DCPROMO. Using dcpromo.exe, we can install and remove Active Directory from a Windows Server 2003 computer.

The computer on which Active Directory Services is installed is called domain controller. It is from this computer that the network administrator manages the domain.

How to setup a DNS server

DNS is a service, which translates computer names (host names) to IP addresses. The DNS server is a machine on which DNS service is installed. The DNS service helps to identify the computers on the network. The Domain Name System (DNS) is the standard name resolution strategy used on Windows Server 2003 systems.

In a practical scenario, when an application is running on the network, the client frequently requests data from the SQL server residing on the server. Whenever a request is issued there is a necessity for the client's identity. The DNS server helps to resolve this identity. At the application level, the DNS server is not involved in the scheme of things. The DNS server operates only at the OS level and helps to open a channel between the server and a client. Once the IP addresses are set and the application is installed, there is nothing to be done on a day-to-day basis. DNS is most commonly associated with the Internet. However, private networks use DNS extensively to resolve computer host names and to locate computers within their local networks. If a DNS solution is not available or installed when you set up domain controller, we won't be able to create the domain. Therefore, you need to have a working DNS server before we can install Active Directory. If not, Active Directory Installation Wizard will offer to create one for you.

How to setup a DHCP Server

In a network, if the number of clients are more, setting up IP addresses manually, is not practical. For such cases, an additional service called the DHCP (Dynamic Host Control Protocol) service is provided by Windows operating system. This service allocates an IP address to each computer when the system boots. The advantage is that, when there are more number of computers on the network, the need to give one unique IP address to each one of the computer is eliminated.

Install and Configure the DHCP service

To implement DHCP, you must install and configure the DHCP service on at least one computer running Windows 2000 or 2003 server within the network. For DHCP to function properly, we must manually allocate the IP address on the server and set up the clients for dynamic address configuration. We will install the DHCP service on the first computer, referred to as SYS1 here on.

Use the Add/Remove programs utility in control panel. Then click the Add Windows Components selection.

1. Choose Networking services, select Dynamic Host Configuration protocol.
2. Click Next, and you will be prompted for the Windows server 2003 source files. Required files will be copied to your hard disk.
3. Click Finish to close the Windows Components Wizard.

How to setup a particular Network Topology

The topology of a network is the pattern used to connect the computers and other devices with the cables or other network medium. There are several basic types of network topologies, or structures in networks. A network can be connected by using any one of the following three topologies:

1.Bus
2.Star
3.Ring

we will discuss the star topology as this is the most common topology used.

Star topology

In a star topology, each computer is connected to the hub using a separate cable. Most of the Ethernet LANs installed today, and many LANs using other protocols as well, use the star topology. Star LANs can use several different cable types, including various types of twisted-pair and fiber optic cable. The unshielded twisted pair (UTP) cables used on most Ethernet LANs are usually installed using a star topology. Each computer connects to the hub with its own cable, the hub propagates all signals entering through its ports out through all of its other ports. The main advantage of the star topology is that each computer has its own dedicated connection to the hub, providing the network a measure of fault tolerance.

Concept of Ip address

An important concept of networking is IP addressing. Currently a 32-bit network address is in use all over the world. This address contains 4 octets i.e, four numbers. Each octet can be any number from 0 to 255. Each octet is separated by a period. IP addresses must be unique for each computer in the network. IP addresses commonly fall within three classes: Class A, Class B, and Class C. Class assignments are based on network size and the availability of IP addresses. Each network device needs a unique IP address. The system administrator, or anyone who coordinates IP address assignment and configuration, should assign this address from the pool of addresses he defines. For example if the network contains 25 computers, he may define it as 1 to 25 (192.168.0.1 to 192.168.0.25). In Class A, the first octet refers to network and the next three octets refer to hosts. In Class B, the first two octets refer to network and the next two octets refer to the host. In Class C, the first three octets refer to the network and the last octet refers to host. The term network and

host are relevant, when we want to be part of the Internet. If the network we are setting up is not a part of the Internet, then we need to use the following IP addresses of Class C, and they are 192.168.0.0 to 192.168.255.255. These addresses are called private addresses. These addresses are designed to be used within an organization. The last octet in this IP address can be of our choice. Note that in DHCP server we have to provide the range of possible numbers, so that DHCP server will allocate a number with in this range automatically. For public addresses to be used on the internet, InterNIC an organization established to monitor and allocate public IP addresses accepts the requests for these IP addresses and allots the same on a first come first serve basis.

SQL Server Database Server installation

After we are done with windows operating system installation and setting up we can now go ahead with sql server installation. The important topics in installation and setting up sql server database are:

How to Secure Databases in Sql Server

Databases have to be secured to ensure that only authorized users can access a SQL Server database. To secure the database the following steps have to be done using the Enterprise Manager. In sql server 2005, there are changes in executing the following steps. If you want to know exactly how it is done in sql server 2005 refer the book titled "Database Programming using visual basic 2005, c# 2005 and sql server 2005".

1.Creating Logins.
2.Adding Logins to Server Roles.
3.Creating Database Roles.
4.Granting Permissions.

1.Creating Logins

In order for users to get to the data provided by SQL Server, they must first log in to SQL Server. This is the first layer of security that is implemented inside SQL Server itself. SQL Server logins control which individuals or group of individuals have permissions to connect to a SQL Server.

There are two basic types of logins. There are logins (windows logins) that are created internally in SQL Server and logins that reference existing windows users or groups. The standard logins are created by the SQL administrator internally in SQL Server and are primarily for non-windows or remote users to use to log in.

2.Adding Logins to Server Roles

As an administrator, we need to allow other people to perform certain administrative activities on the SQL Server. By default, individuals with login access to SQL Server have no implied privileges on the server. If you require specific logins be allowed to perform certain administrative tasks on the server, you need to give users the permissions to do so.

Server roles are basically groups that exist at the server level. The server roles are built into SQL Server and have specific permissions pre-assigned to them. By adding logins to these Server Roles, you allow those logins to perform the actions for which the role has permissions. For example, adding a login to the dbcreator server role allows those users to create databases in SQL Server. The set of roles that exist at the server level and the permissions assigned to them are all predefined. You cannot create your own server roles, nor can you change their permissions. Because of these restrictions, the server roles are called fixed server roles.

3.Creating Database Roles

In SQL Server, groups are called roles. Server roles exist at the server level and allow their members to perform server wide operations. SQL server also provides a number of built-in roles for each database it contains. The built-in database roles, better known as fixed database roles, provide a convenient mechanism for assigning special database permissions to users. SQL Server does allow you to create your own database roles.

4.Granting Permissions

Permissions need to be assigned for users to be able to execute statements and interact with the objects in a database. Permissions can be assigned using Enterprise Manager. In the enterprise manager, Expand the Databases, select the Northwind database and Click the Tables object in the Enterprise manager tree. Then all the tables will be displayed in the right pane. Click on the Categories table, properties dialog will be displayed. Click on the Permissions button, and then the object properties dialog box will be displayed. A checked box for permission implies that the permission is granted. A box with an X in it implies that the permission has been denied, and a cleared check box means that the permission is revoked.

Understanding the Windows Authentication Mode

Windows authentication is generally preferred because it provides an optimal level of integration with Windows 2003 server. User and group accounts from Windows are granted or denied access to SQL Server. Windows 2003 authenticates the user when the user logs on to the network. Because the password is authenticated at network login, SQL Server does not need to know or verify the password of a user. Windows Authentication provides the following advantages over SQL Server Authentication.

Windows Authentication can grant group accounts to access SQL Server, thus minimizing the over head of login administration.
Users are authenticated by Windows 2003, resulting in a secure authentication over a network.
Users could be able to use the same user credentials for network and database access.
Audited events can be tracked to a network user.

Understanding the SQL Authentication Mode

SQL Server Authentication is preferred in the following scenarios.

1. The user is not logging into a Windows domain.
2. Your network does not have a Windows 2003 domain.
3. It is not feasible to manage all Internet users on the windows domain.
4. You prefer to manage them separately from your normal Windows domain administration.
5. An application is acquired from a vendor that requires SQL Server Authentication.

How to Create Logins in Sql Server

To add a Standard SQL Login using the Enterprise Manager, follow these steps.

a. Select your server in the Enterprise Manager tree.
b. Expand the Security node, and select logins.
c. Right-click Logins, and select New Logins from the pop-up menu.

d. Ensure that the SQL Server Authentication option is selected.
e. Enter the name of the SQL login in the Name field.
f. Select the default database for the user.
g. Click OK.
h. Confirm the password for the login.

Understanding SQL Server Security

To develop a client application that establishes a connection to a data source, we must plan the way in which the connection is made. This includes determining the security mode of the designated data source, and whether it requires a user ID and Password. Security is necessary to protect the information contained in the database. We usually implement one of two primary security options.

1. Standard security mode.
2. Integrated security mode.

1. Standard security mode

Standard security mode is the default security mode for SQL Server. Standard mode uses the SQL Server security model for every connection to the database. It supports non-trusted environments, such as the internet. Note that users will not necessarily first connect to a Microsoft Windows 2003 or NT server for authentication. SQL Server will perform its own authentication in this situation. The system administrator or database owner can create userIDs, user names, and groups for each database on the server. When this mode is in use, the user must enter a user ID and password that has been established for the database.

2. Integrated security mode

Integrated security mode allows SQL Server to use Microsoft Windows 2003 or Windows NT authentication mechanisms to validate all connections to the database. We use integrated security in network environments in which all clients support trusted connections. Integrated security allows applications to take advantage of Windows server security capabilities. With integrated security, user maintain a single user ID and password for both Windows 2003 and SQL Server. If SQL Server is using Integrated Security, the values of user ID and password are ignored.

Sql server uses the service security account to access windows resources. The service security account is the logon account that SQL Server uses to access Windows resources. The service security account is not a login for users connecting to SQL server.

Creating a Dedicated Service Security Account

Two options exist for the service security account. The first option is local system account and the second option is dedicated domain user account.

The local system account is a Windows 2003 operating system account with full administrator rights on the local computer. We use this account to install SQL server where we need not integrate SQL Server with other server applications such as Exchange server.

The recommended option in production environments is to use a dedicated domain user account. The preferred way to use a dedicated domain user account is to create an account in a Windows Active

Directory domain that can be referenced by all the computers involved in a domain. We recommend creating a specific account just for SQL Server rather than share a general network system administrator account. This will help reduce the chance that a network-system administrator will one day delete the account or change the password, causing SQL Server to fail.

We use Active Directory Users and Computers to create and configure a dedicated Windows 2003 user account. The following are the steps to create a service security account.

1. Ensure that, you are logged on to the domain controller as Administrator.
2. Click Start->Programs->Administrative tools and then click Active Directory users and computers. The Active Directory users and computers screen appears.
3. In the console tree, expand VKINFOTEK.com. Right click users, point to new and then click user.
4. The New Object – User dialog box appears.
5. In the First name text box, type SQL Service. In the User logon name text box, type SQL Service, and then click next.
6. Type the password in the Password Text Box and Confirm Password text box and select the Password never expires check box, and then click Next.
7. Click the Finish button.
8. In the console tree, click users. The details pane displays a list of all users in the Users container. Verify that the SQL service domain user account has been created and close Active directory users and computers.

In a production setting data entered by the user is stored in a sql server database tables. These tables cab be created using designer or using the scripts. Using scripts is a more professional way of designing the database. Let us now see how it is done.

Creating Tables with SQL Scripts

If you are developing a database for mass deployment or repeatable installations, the benefits of developing the database schema in scripts become obvious, as listed below.

1. All the code is in one location. Working with SQL scripts is similar to developing an application with Visual Basic or C#.
2. The most current version of the database may be installed without running change scripts or restoring a backup.

All the tables used in the application series books are created using scripts.

Drawbacks

The T-SQL commands may be unfamiliar and the size of the script may become overwhelming.
In a situation where foreign-key constraints are embedded within the table, the table-creation order is very strict. If the constraints are applied after the tables are created, the table-creation order is no longer a problem; however, the foreign keys are distanced from the tables in the script.

The following CREATE TABLE DDL command creates the Customer table. The table name, including the name of the owner (dbo), is provided, followed by the table's columns. The final code directs SQL Server to create the table on the primary filegroup. Apart from the columns, the only information you normally supply when creating a table is the name.

```
CREATE TABLE dbo.Customer (
CustID INT  NOT NULL PRIMARY KEY NONCLUSTERED,
CustName VARCHAR(30) UNIQUE NOT NULL,
```

```
City VARCHAR(50) NOT NULL
)
ON [Primary]
```

Creating Keys

Enforcing data integrity ensures that the data in the database is valid and correct. Keys play an important role in maintaining data integrity.

Creating Primary keys

The relational database depends on the primary key. The uses of the primary keys are:

1. To uniquely identify the row.
2. To serve as a useful object for a foreign key.
3. Primary keys can be single columns for fast joins and where clauses.
4. Primary keys never need updating.
5. Note that Primary keys should not contain data that dynamically change, such as a timestamp column, a date-created column, or a date-updated column.

Setting a column, or columns, as the primary key in Enterprise Manager is as simple as selecting the column and clicking the primary-key toolbar button. Enterprise manager creates primary keys with clustered indexes. This is a poor index choice and waste of the one clustered index available for a table.

In the following code, declaring the primary-key constraint in the Create Table statement is shown.

```
CREATE TABLE dbo.Customer (
CustID INT  NOT NULL PRIMARY KEY NONCLUSTERED,
CustName VARCHAR(30) UNIQUE NOT NULL,
City VARCHAR(50) NOT NULL
)
ON [Primary]
```

Two data types are excellent for primary keys: Identity columns and unique identifier columns.

Using Identity columns

By far the most popular method for building primary keys involves using an identity column. Like an auto-number column or sequence column on other databases, the identity column generates consecutive integers as new rows are inserted into the database.

Advantages of using Identity column for primary keys:

1. Integers are easier to manually recognize and edit than GUIDs.
2. Integers are small and fast
3. An identity column used as a primary key with a clustered index (a common, but poor practice) may be extremely fast when retrieving a single row with a single user. However, that configuration will cause lock-contention hot spots on the database.

Identity – column values are created by SQL Server as the row is being inserted, as shown below.

```
CREATE TABLE dbo.Customer (
CustID INT  IDENTITY NOT NULL PRIMARY KEY NONCLUSTERED,
```