OVERVIEW OF DIFFERENT APPLICATION SERVER MODELS

Before you start

Objectives: learn what is application server, what is thin-client and what is fat-client, and about different types of application server models.

Prerequisites: no prerequisites.

Key terms: application, server, model, network, client, type, hardware, time, multiple, physical, software

What is Application Server

The term "application server" identifies services that are running on the server hardware. An application server provides network applications that users can use. Examples of that are various **collaborative** or groupware applications, such as e-mail, group calendars and group tasks. Those applications enables us to organize time and appointments for multiple people inside of our organization. We run those applications from our application server. The network application that users are using on their local workstation actually runs off the network server.

Another commonly used network applications are **databases**. Databases are used to store a lot of data which can then be accessed and create reports based on that data. In this case we run a database software on our server, which can be accessed by other users from other workstations on our network. The users can connect to the database, run searches, enter new information, and generate reports. Remember that the database resides on the server, not on individual workstations. Because of that fact the server is functioning as an application server.

Another common applications that can be served are **web-based applications**. To provide web applications we use some technology to build web pages either statically or dynamically, and then serve those pages using web-server software. The clients can then connect to the server using a web browser and request a certain web page. The application server dynamically generates content and returns it back to the client in a form of a web page.

Thin-client versus Fat-client

With an application server most of the processing is being made by the server. The workstations usually simply display information from the server, but this can vary based on the type of application. In general there are two different application models. The first one is called the **thin client model**. By using this type of model, the workstation does no work whatsoever, or it does very little work when compared to the server. In this model 99% (or more) of the work is done by the server. On the other hand we have a **fat client model**. In this type of model we have applications that require more processing time on the end-user workstation. A part of the processing is still done on the application server, but the part of the work is also done by the client workstation. Processing on workstation can be 50% or even more.

Different Types of Application Servers (Models)

We have three different types of application servers, which are actually the application server models that we should be familiar with.

The first one is called a **dedicated application server**. With this type of application server we are talking about one physical hardware server which is running one single network application. Some applications will actually require this. The benefit of using a dedicated application server is great performance, because the server hardware is serving a single application. All CPU and I/O resources are dedicated to providing a single service. However, this model is more expensive because it will force us to to have more physical servers installed for other purposes. That's because organizations typically have to use more than one network application in their environment, and in that case we will need to purchase additional physical servers. Another issue depends on how heavily the network applications is being used. Depending on application usage, this model may result in unused capacity if the application is not often used. In that case the server is idle most of the time, which is not very efficient. However, most of the time this will not be an issue since dedicated servers are used for applications which will be used a lot.

The next type of application server model is a **distributed application server**. With this model we still have one network application which we run on many different servers concurrently. So, we use multiple servers to provide the service. Many Web-based applications use a distributed model. More servers allow us to configure fault tolerance by using special software. If one of the servers goes down, other servers can take its place without any down time, and the end-user won't even know that something happened to our server. More servers also mean scalability which means that we can balance the load between different servers by using a special software. This can be done in various ways, but for now it is only important to remember that different servers can share the load, which enables us to increase the number of users that can use the network application. End-users will see all this as a single service running on a single server, but we know that it's actually multiple servers running one application. The downside of this type of model is the cost of implementation because it requires more hardware. Also we might have to pay extra money for the application to be able to run on more machines instead of one. We also might have to purchase some type of clustering software or load-balancing software to manage the load among different servers and provide fault tolerance feature.

The last model we should know about is the **P2P or Peer-to-peer model**. When talking about computer networks, we already know that we can configure a peer-to-peer type of computer network. In this type of network we have workstations that provide services on the network and in the same time consume services on the network. So, the systems are a server and a client at the same time as opposed to client/server network where servers provide services and clients consume services. In P2P application model we have application that runs on multiple systems which contribute with their resources. The applications could be running as a client or running as a server for other instances of an application. The key advantage of a P2P application is that it uses server hardware very efficiently. If we have many different server systems, the peer-to-peer application model can balance the load more effectively when compared to distributed application server model, and provide the best performance possible. This is accomplished by analyzing the servers and determining how to load balance. However, the peer-to-peer application model is more complex to set up.

Remember

Application server model is a way in which we make our network applications available to client workstations. In dedicated application server model we have one physical hardware server and one network application. In distributed application server model we have one application running on many servers concurrently. A peer-to-peer (P2P) application server model multiple instances of an application can be running on multiple systems. In P2P each application instance can act as a server application or client application.

Source : http://www.utilizewindows.com/server/basics/372-overview-of-different-application-server-models