A **virtual directory** is a **directory** name that you specify in IIS and map to physical **directory** on a local server's hard drive or a **directory** on another server (remote server). You can use Internet Information Services Manager to create a **virtual directory** for an **ASP**.**NET** Web application that is hosted in IIS.

Where are IIS virtual directories stored?

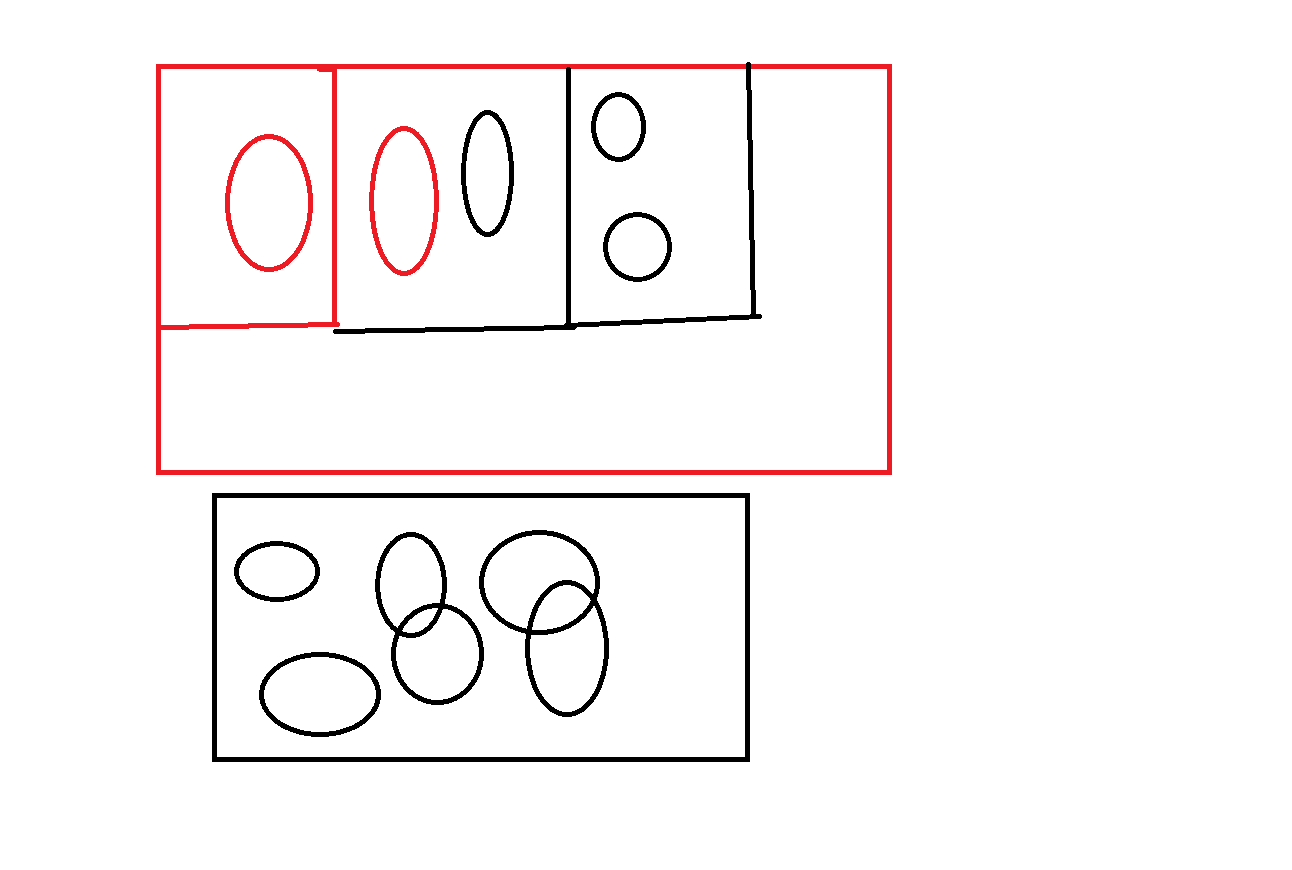
The entire **IIS** configuration for IIS6 is in c:\windows\system32\inetsrv\metabase. xml. Just search for your site name and you should jump to the web section (move past the FTP section if you stop there first). Get the ID from the path and then keep searching by {id}\ until you find the **virtual directory** details.

Internet Information Services (**IIS**) is a flexible, general-purpose web server from Microsoft that runs on Windows systems to serve requested HTML pages or files. An **IIS** web server accepts requests from remote client computers and returns the appropriate response.

Application Pool

An **application pool** serves as a container for your **applications** in **IIS**. It's a collection of one or more URLs that can be served by a worker process, and it provides isolation: **applications** that run on one **application pool** are in no way affected by other **applications** that run on different **application pools**

**[What is application pool in IIS in layman terms?](https://www.quora.com/What-is-application-pool-in-IIS-in-layman-terms" \t "_blank)**

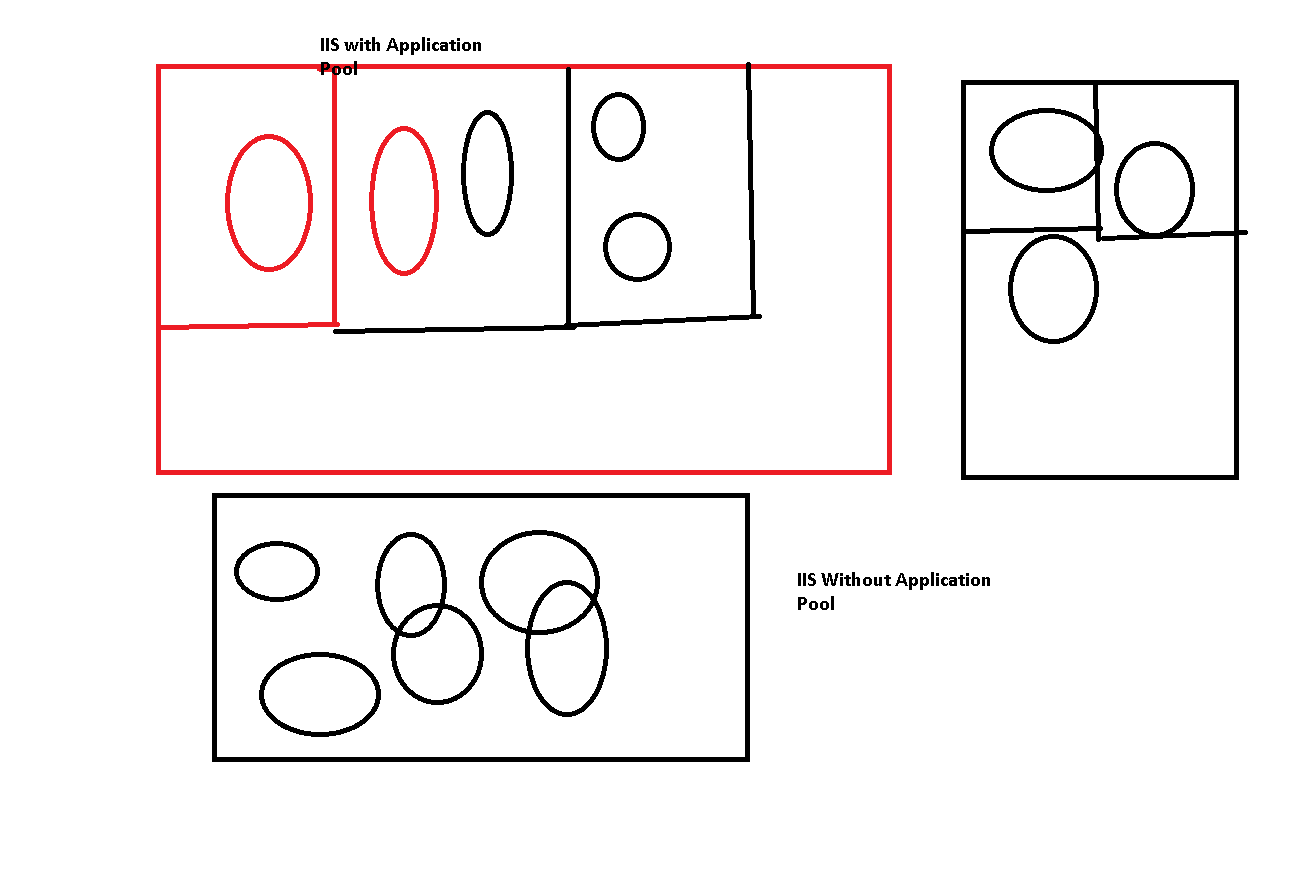
****

At the beginning, IIS was a single process; all web sites on a computer worked under that single process. It was not a problem when the sites had static contents. But when the programs run on the sites to make them more dynamic we had many problems; because IIS was a quality program but it couldn’t dictate the web programs what to do and not to do. The result was that a single program on the computer could crash the IIS! It means the faulty program can crash all the web sites.

So, there must be a mechanism to isolate web sites and their applications from each other so a faulty program could crash only itself. This mechanism is the application pool. Every web site on IIS has an application pool. This application pool need not to be unique: More than one site can share a single app pool but frequently they have their own application pools.

**What is IIS Root Directory**

Open **IIS** Manager and under Sites there should be **IIS** websites and under it you will probably **see** Default website. You can **see** the **root directory** with Right click/Settings. Copy your website files under this **root directory**.



To Access Database

You'll need to change the application pool identity to NETWORK SERVICE as well. Fire up Internet Information Services (IIS) Manager, drill down to Application Pools, and select the Application Pool that you're using. (Not sure? Drill down to your virtual directory under Sites on the left side and then click Basic Settings on the right.) Then click your Application Pool, click Advanced Settings, and then change Identity under Process Model to NetworkService.

In SQL Server Management Studio, you'll need to add the NT AUTHORITY\NETWORK SERVICE user under Security for your database,

