Software choices to implement a remote database client/server network setup

Fortune Walla

August 16, 2015

Posted on Thursday, May 8th, 2014 at 2:25 am

Interacting with a database on a network from statistical/reporting software like MS Excel, MS Access, SAS, R, Python, etc... is a great way to learn how data might be retrieved, analysed & stored remotely. Web scripting languages like ASPX, PHP also allow remote database interactions for data analysis using Web based applications.

However, a local database with local software using only individual local files pretty much provides the same experience as using them on a network. But the aim here is to simulate a corporate/research environment where all the software & data resources are spread throughout the network. The aim is not to become an expert in the technologies but to know just enough to use them to do data science/analytics.

Hence the philosophy is to use minimum hardware/software resources to be able to study & learn quickly and efficiently. The problem is that each person has a different configuration of client/server and hardware/software components & usually one person's solution will not necessarily work for the other. The choices available to you will be different from the choices presented here.

Server decision process:

- Server database: MS SQL Server (MSSQL) vs. Mysql Database (mysql)

  Decision: MSSQL

  Reason: MSSQL has built-in data mining through SQL Server Analysis Services (SSAS). Also MSSQL2008 offers more data mining options when compared to MSSQL2005. There is more information online about MSSQL data mining features than about mysql data mining features. Furthermore, MS Excel can perform data mining directly on spreadsheets using SSAS via an add-in

- Server OS: Windows XP (winxp) vs. Windows Vista/7 (win7)

  Decision: win7

  Reason: win7 has similar features to Windows Server 2008 & handles networking better than winxp. Winxp would be sufficient but one needs to install additional software components such as dotnet packages, VC++ run-time packages, deal with permissions/networking issues etc...

Once you have the server & software setup for networking & remote access, almost any hardware/software combination can be used as a client.

Client decision process:

- Client OS: Windows vs. GNU/Linux

Decision: Windows

Reason: MS Excel and MS Access are used widely in the business/research world. Hence Windows XP was chosen. Also MS Excel supports many add-ins for statistical analysis, data mining & visualization. Although software such as R, Python, WEKA etc... would integrate better with GNU/Linux.

- Client web server: MS Internet Information Services (IIS) vs. Apache Web Server (apache).

  Decision: IIS

  Reason: A student of data science/analytics does not need advanced features of a web server. A basic version of IIS comes built-in with winxp. I am guessing it is enough for the purpose of learning. Apache is robust but requires more configuration than IIS.

- Client web language: ASPX vs. PHP

  Decision:PHP

  Reason: PHP is easy to configure for IIS & light on system resources. ASPX would be the preferred choice for use with IIS and MSSQL, but requires more configuration & systems resources for Visual Studio 2010 development environment. Students can look at the PHPStats project https://github.com/mcordingley/PHPStats for using statistical functions to do data analysis on the web.

- OS for Web Server: Client OS vs. Server OS

  Decision: Client OS

  Reason: Decision to install IIS/PHP on Client OS was primarily done for two reasons i) To isolate the database from any instability that might result from the Web application environment ii) To simulate an "Internet" where the database & web server reside on different machines.

  Misc. client software: MS Excel, MS Access, R, Python. It is advisable to try the database access features of many software for learning & practice.

Summary:

Final server setup: SQL Server 2008 on Windows 7.

Final client setup: PHP using IIS on Windows XP.

The main idea is to simulate a corporate/research data science/analytics environment where data & software resources are spread throughout the network. The design choices are made to use minimum resources. This enables students to learn the basics efficiently without having to worry about the advanced features. The implementation of these choices will be discussed in another post.