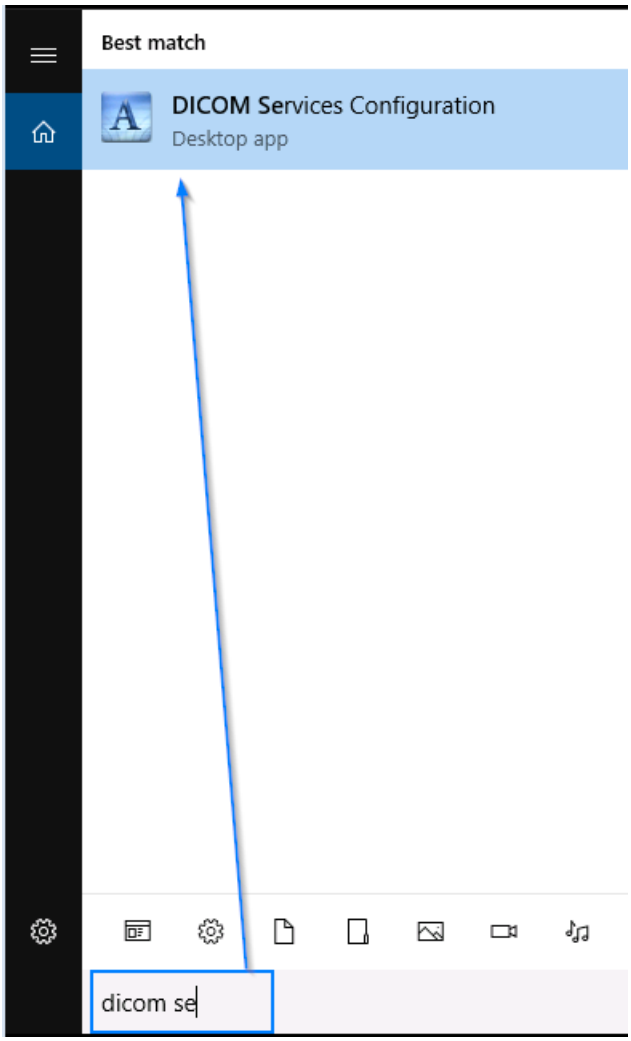
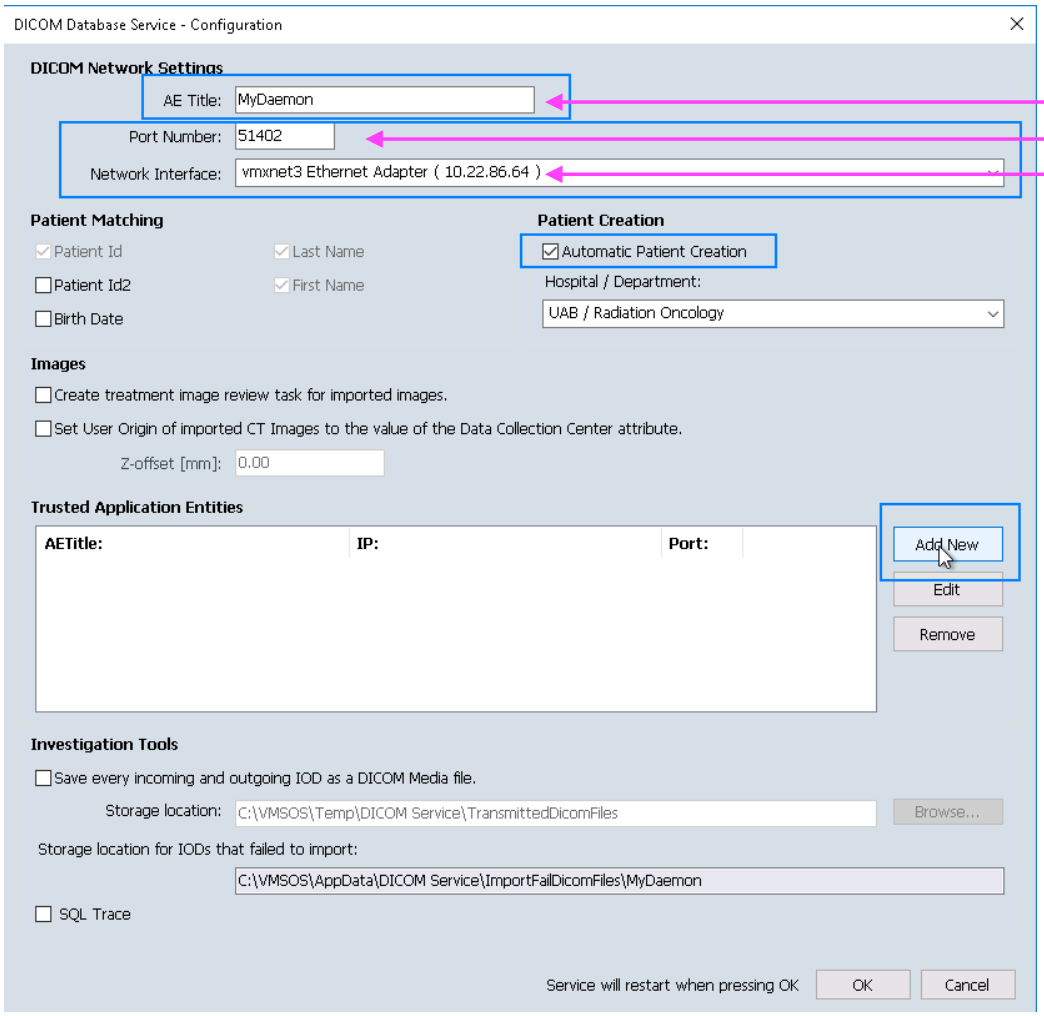


4.2.2 Spawning a Daemon

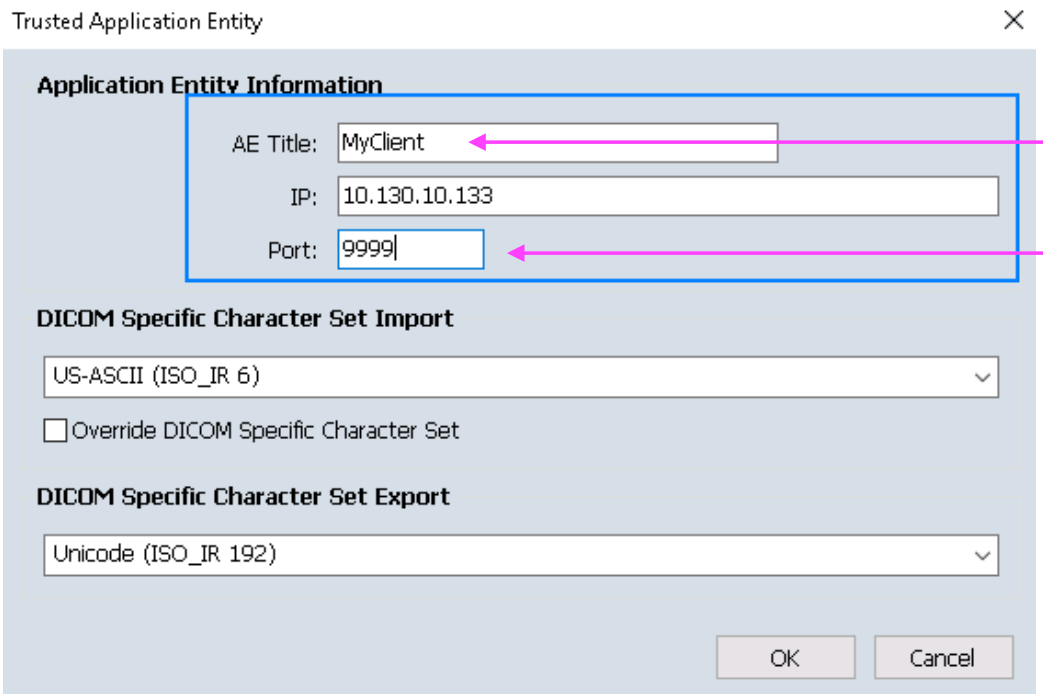
1. Select **Add Service**
2. Select **Database Service**
3. Set the **AE Title**, **port number**, and record the **IP address** from the ethernet adapter.
4. **Check** Automatic Patient Creation
5. Select **Add New** (trusted entity)



remember
for my program
DaemonTitle
DaemonPort
DaemonIP

4.2.3 Adding a Trusted Entity

This next step is site specific. You will need to whitelist some computers so that they can talk to the daemon. For each computer that will need to communicate with the daemon, you will have to add a new trusted entity. The AE title just needs to be a unique name for the DICOM nodes. The IP address will be the address for the computer from which you will be calling (probably where Visual Studio is installed). Remember the port that you assign here. It will be important for some operations.



We are now ready to start communicating. The Varian daemon can perform several useful operations for a client. Each of the operations is explored in the following sections.

4.3 DICOM Language Basics

The exact bitwise operations of the protocol are beyond the scope of this chapter. Instead we will focus on high level communication concepts more relevant to the DICOM library user. If instead, you were trying to develop a library or implement a DICOM call from scratch, I recommend the book Digital Imaging and Communications in Medicine (DICOM): A Practical Introduction and Survival Guide by Pianykh. Additionally, the DICOM transport layer documentation (part 8) provides very detailed information. For the purposes of learning DICOM communication using an existing library, we will continue to use Evil DICOM. For details on setting up a project with Evil DICOM, see the previous chapter. The main DICOM calls we are interested in here are the following:

- C-ECHO (DICOM "ping")
- C-FIND (DICOM search)
- C-MOVE (DICOM move and storage)
- C-STORE (DICOM storage)

4.3.1 C-ECHO

The first network operation is the simplest. When starting to set up a DICOM network, the DICOM creators designed a mechanism equivalent to the ping command in Windows. Successful execution of the following command ensures that 1) the DICOM node is up and running and 2) it is allowed to communicate with the client.