CAREWare has a four-tier architecture and the only way to safely bypass the CAREWare client and insert data into CAREWare from other systems is to learn to use the public interface the business tier provides. The standard CAREWare 6 application uses the same business tier interface that you will be using, so you should be able to write some powerful third party tools. The way to make these tools is to write your own programming code that uses the business tier functions via the HTTP Server, and before you can do this you will need to learn a little about how CAREWare uses the interface. This example is a good place to start.

The CAREWare Business Tier is a windows service that is installed on a computer that listens and responds to requests from the CAREWare HTTP Server. You can think of it as a communication hub between any number of client applications and the underlying data in the SQL Server. To connect to it you will need to use our CW6ApiClient and supply the correct IP Address, URL or Machine Name and port number it is listening on, and have a valid CAREWare username and password, but we are getting ahead of ourselves.

In each installation of CAREWare there is only one business tier and one HTTP Server, if your organization is large chances are it is on a server somewhere. If you just installed all tiers on your PC then you can find them (CAREWare Business Tier Service and CAREWare HTTP Service) in your services list. The approach I am going to take here is to build a new DotNet project from scratch, reference our CW6ApiClient and ezjson projects or dlls and use it to connect, login, and use the CAREWare public interface. To find out the kind of things you can do with the CW6ApiClient you can reference the following document here.

In this example we make a tool that adds a fake client, adds a CD4 Count to their record, downloads the client's lab history, shows how to inspect the returned object, deletes the lab, deletes the client, and then logs out. Before you try to connect to the business tier through your programming code, first make sure you can connect and login through your browser.

Once you have verified you can login through the browser, you can proceed to make your project.

Getting Started

Download and unzip the sample application here and either make a new DotNet Project or just use the sample. In the sample application, I am using a VB.Net Console project, but you can just as easily connect using any DotNet language with a Windows Forms or Windows Service project.

If you opt to make a new project, from the Microsoft Development Environment Click on *File* and choose *New Project*.

Select the application type, name the project, and click *OK*.

Reference the CW6ApiClient and ezjson projects or DLLs from the unzipped sample application.

If you opt to use the sample application, go into the unzipped sample and open Cw6ApiDemo.sln.

We have made a CWApiSession class that simplifies connecting to the CAREWare business tier. You can check out the source code for it if you are interested in the details. The CWApiSession class is contained in the CW6ApiClient project.

The sample application connects to the business tier using the following code snippet:

```
Dim useTLS as Boolean = False

Dim serverName as String = "localhost"

Dim providerName as String = "default"

Dim sess as New CWApiSession(useTLS, serverName)

CWApi.Session.Login(sess, "cwtemp", "TEMPCW100", providerName)
```

This connects to the default provider on the local business tier without TLS using the default login and password.

Summation

The CW6Api provides operations for many of the functions available through the standard application but not all of them. The plan is for it to grow over time. If you find that there is something that you would like added, submit a request through the CAREWare Help Desk at cwhelp@jprog.com.