

Using Windows Services for Clinical Research

Dr A. Philips, Specialist Trainee; Dr I. Wrench, Consultant Anaesthetist; Dr R. Freeman, Consultant Anaesthetist

Anaesthetic Department, Sheffield Teaching Hospitals NHSF Trust, Sheffield, England

A typical PC will run upwards of 30 programs even when idle. Most are classed as Windows Services because they start up automatically, remain active while the machine runs, and require no user interaction. While there are probably few people in the world who know what these do specifically, in general terms they wait for action from devices such as keyboard, mouse, disc, network, etc., passing that data on after processing, or they carry out some action on a timed basis. Microsoft Visual Studio 2005 included a template for writing a Windows Service.¹ We had looked at this for an application for checking patient information against PAS, but found it overly complex in that role.

Datex AS/3 and CS/3 monitors have been used widely in STH for over 10 years, and some years ago we developed applications that download data from them on a timed basis for research and other purposes. These devices have an RS232 port which will connect to the same on a PC via a null modem cable with appropriate hardware settings. The PC sends the Datex device a string, which causes it to transmit the values in its averaging database. If the PC waits 2 seconds and examines the contents in its own RS232 receive memory it will find this data as another string, which it can decode and save, or save for later decoding.

We were asked to provide this application for a research project looking at the quality of pre-oxygenation for caesarean section. The application required a user to be logged on to the PC and to start it acquiring data. This research project needed the PC to continually look at the data from the monitor without user interaction. Though we logged on to the PC and started the data acquisition, the PC shut itself down after a variable period because of the lack of user interaction. This application miserably failed the research goal.

We admit we hoped this would go away, but our researchers are persistent, and as they came up with ever more worrying schemes for capturing the data, we had to look at the problem again. An application that carries out its function on a regular timed basis, without user interaction, or indeed a user session in progress, is a Windows Service, so we copied and pasted our existing application into the appropriate Visual Studio 2008 template. These Net framework applications are aggressive with objects that seem to do nothing, so we had problems with our service being garbage collected. A Windows Service application also has to be deployed properly, otherwise it will not work as a service.² This deployment proved more complicated than the development of the service. Nevertheless we ended up with a PC that, once running, automatically started up a Windows Service that attempted to interrogate a Datex monitor via its RS232 port every 17 seconds, saving any result to a database. The PC required no keyboard, mouse, monitor, or user session. The database was downloaded periodically onto a laptop connected via a crossover network cable so that the PC was not disturbed.

This system successfully collected the required data continuously over a series of weeks, the consequent results to be presented elsewhere. We suggest that consideration be given to using a Windows-Service-type development for applications that require timed functions without user interaction.

REFERENCES

1. [http://msdn.microsoft.com/en-us/library/9k985bc9\(VS.80\).aspx](http://msdn.microsoft.com/en-us/library/9k985bc9(VS.80).aspx)
2. [http://msdn.microsoft.com/en-us/library/ddhy0byf\(VS.80\).aspx](http://msdn.microsoft.com/en-us/library/ddhy0byf(VS.80).aspx)