Literature Review

Windows Services

Definition

Microsoft Windows services, also known as NT services, help user to create long-running executable applications running in the background as long as the Windows session is live. Windows services can be started, paused and stopped automatically whenever the PC boots up or otherwise. Since these services do not operate with a user interface, it makes them ideal to run on servers or mainframes. These services act as long running scripts executing commands. Windows services can also be run on a specific User account. They can be managed by employing Windows Service Manager. Since they run on the Operating System level, the services must have proper security clearances.

Basic Features

* + All services must be installed on the System or User level before being started.
  + Windows Services run in a different security context and hence cannot interfere with User’s interface.
  + .NET framework does not allow the services to interact with the interactive sessions of the User.
  + A service can exist in one of three basic states: running, paused or stopped. A service can exist in its running state indefinitely until it is either stopped or paused or until the computer shuts down.

Types

There are two broad categories of Windows Services. Services that are the only service in a process are assigned Win32OwnProcess whereas the ones that share a process with one or more services are assigned Win32ShareProcess.

Architecture

Any Windows service application inherits from System.ServiceProcess.ServiceBase class. The service can utilize the following methods in order to articulate the basic functionality of a background process:

* OnStart
  + Actions to be taken when the service starts. It can either be a call to another method or can contain the business logic of the application.
* OnPause
  + Actions to be taken when the service is paused either manually or programmatically.
* OnStop
  + Actions to be taken when the service is stopped either manually or programmatically.
* OnContinue
  + Actions to be taken when the service restarts from after being paused.
* OnShutdown
  + Actions to be taken when the PC is shutting down, if the service is running then.
* OnCustomCommand
  + Actions to be taken when the service returns a custom command.
* OnPowerEvent
  + Actions to be taken when a power management event is received, such as a low battery or suspended operation.

The ServiceBase class’s Run method must also be called in order to initiate the process of the service. In order to be able to install the service and furthermore run it, the service class must also inherit System.ServiceProcess.ServiceProcessInstaller. By overriding its methods the program can add executable classes that are used further. Services can be run directly from this class by calling appropriate post installation event handlers.

The service is installed onto the system by a .NET utility called InstallUtil.exe . This executable file uses the classes created above and loads the service onto the system. If not left to be manual, the service starts after the installation as coded in the post installation event handers.

Windows Comm