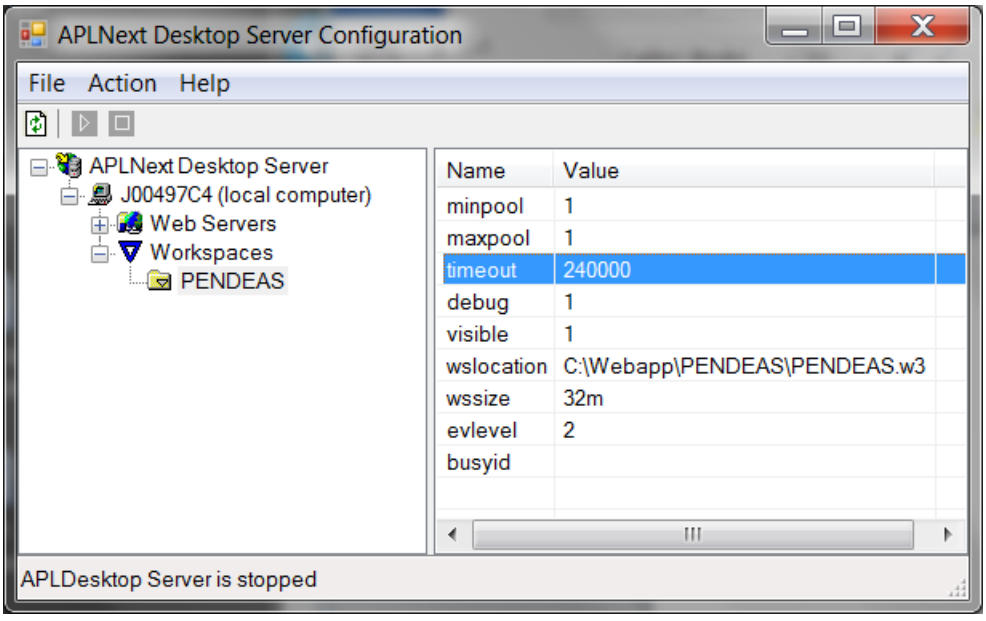
APLNext WebServices
Timeout and Debug Settings

APLNext WebServices is software based on the Microsoft .Net Framework that can be used to expose an APL+Win function as a web service. As a web service an APL+Win function can be accessed by a remote client such as an Internet browser, another web server or any application system that can connect to the public Internet or a private intranet.

To expose an APL+Win function as a web service, APLNext WebServices software is used to associate an APL+Win function with a virtual path on the host machine, and optionally a specific web server port. The APL+Win workspace containing the APL+Win function is also associated with APLNext WebServices. This workspace association includes the 'timeout' and the 'debug' settings. The association of an APL+Win function, its containing workspace and the web server's virtual path and optional port is done via an APLNext WebServices xml-format configuration file. APLNext WebServices includes an administration tool to create, configure and save the configuration file.

Timeout Setting: Limits the duration of the client/server connection
The value of the timeout setting is in milliseconds. For example in the APLNext WebServices Administration Tool screen capture below, the timeout setting value is 4 minutes (or 240 seconds or 240,000 milliseconds).



Generally a timeout of more than 30 seconds is considered very large for a web service which has a human interface, such as a browser-based application system. However since APLNext WebServices can also be used for server-to-server communications and batch processing of multiple transactions, the timeout setting value must set by the APL+Win programmer considering the process to be performed.

APLNext WebServices will end the connection between the APLNext WebServices client and server when the execution time of the APL+Win function associated with the virtual path exceeds the value of the timeout setting value. The connection and transmission timespans between client and server are not included in the timeout timespan.

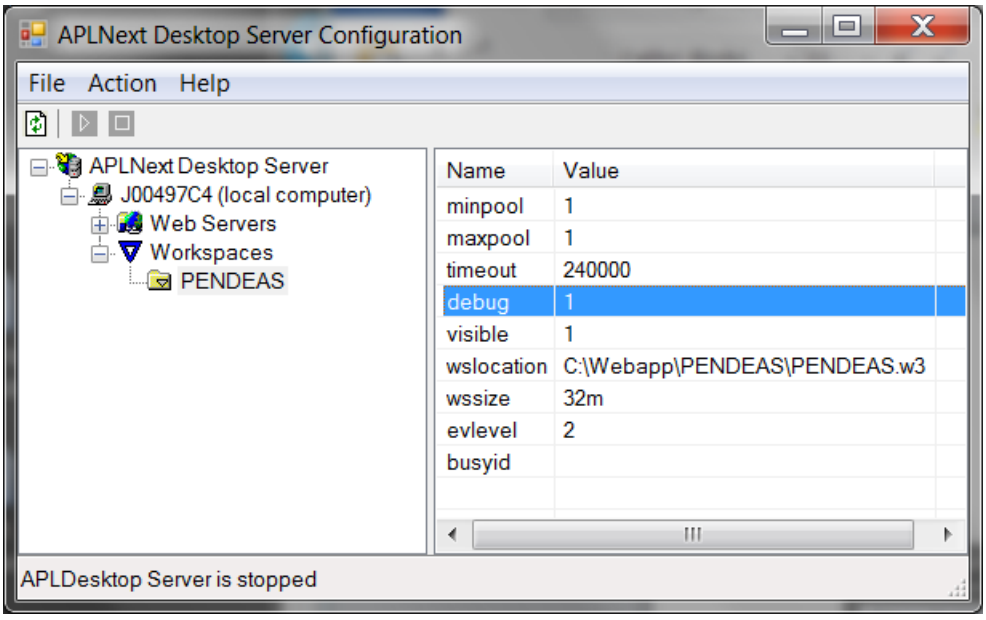
For example suppose that the APL+Win function 'FOO' is exposed as a web service using APLNext WebServices via the virtual path "http://www.mydomainname:9000/myvirtualpath/foo?arg1=val1&..." Further suppose that the APLNext WebServices timeout setting value for the workspace containing the 'FOO' function is 15000 milliseconds (15 seconds). If a client request to run the 'FOO' function is received by APLNext WebServices and the time to complete the execution of that function exceeds 15 seconds, the connection between the client and APLNext WebServices will be ended before the function 'FOO' has completed operation and returned its result back to the client.

The purpose of the APLNext WebServices timeout setting is to protect the server. Set the timeout to prevent degraded server performance or reduced availability to the majority of clients due to an anomalous client request which might otherwise consume excessive server resources. The optimal setting must be determined by the APL+Win programmer based on the experience of timing the APL+Win function exposed as a web service.

Using timeout setting value which is very large ('just to be safe') actually creates a security risk for a production server because a malicious client could make requests which effectively consume all available server resources thereby denying legitimate clients the use of that server.

Debug Setting: Overrides the Timeout Setting

The value of the debug setting is 0 indicating that it does not apply or 1 indicating that it does apply. For example in the APLNext WebServices Administration Tool screen capture below, the debug setting value is 1.



Generally a debug setting value of 1 is transiently used by the APL+Win programmer to override the timeout setting on an APLNext WebServices web server for the purpose of testing and perfecting an APL+Win function exposed as a web service.

When the value of the debug setting is 1, APLNext WebServices ignores the timeout setting value and instead does not limit the timespan of the connection between the client and the APLNext WebServices web server. This means that the APL+Win programmer can use a test client to request execution of the APL+Win function via APLNext WebServices and then examine the action of that APL+Win function on the server side. This server side examination can employ the traditional and very effective APL+Win interactive debugging, tracing, program stops and code walker tools to review global and local variables, arguments and results.

It is recommended that the debug setting value of 0 and an appropriate timeout setting value be used with an APLNext WebServices web server in production mode to prevent the reduction of server performance or availability by malicious clients.