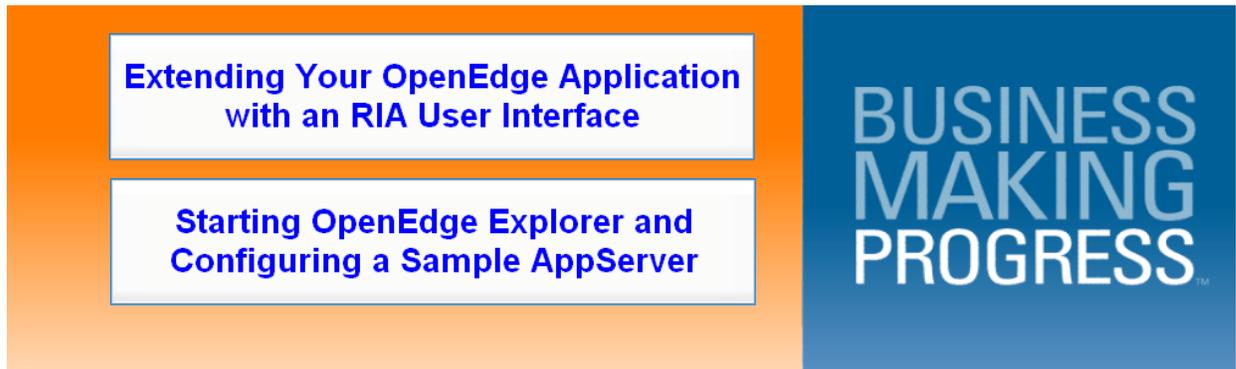STARTING OE EXPLORER AND CONFIGURING A SAMPLE APPSERVER

John Sadd
Fellow and OpenEdge Evangelist
Document Version 1.0
August 2010

Extending Your OpenEdge Application with an RIA User Interface

Starting OpenEdge Explorer and Configuring a Sample AppServer

BUSINESS MAKING PROGRESS™

John Sadd

Progress **OpenEdge**®

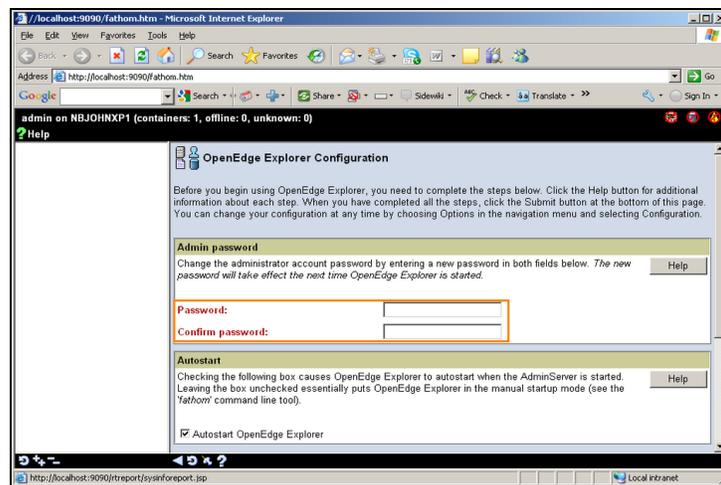PROGRESS SOFTWARE

PROGRESS SOFTWARE

DISCLAIMER

Certain portions of this document contain information about Progress Software Corporation's plans for future product development and overall business strategies. Such information is proprietary and confidential to Progress Software Corporation and may be used by you solely in accordance with the terms and conditions specified in the PSDN Online (http://www.psdn.com) Terms of Use (http://psdn.progress.com/terms/index.ssp). Progress Software Corporation reserves the right, in its sole discretion, to modify or abandon without notice any of the plans described herein pertaining to future development and/or business development strategies. Any reference to third party software and/or features is intended for illustration purposes only. Progress Software Corporation does not endorse or sponsor such third parties or software.
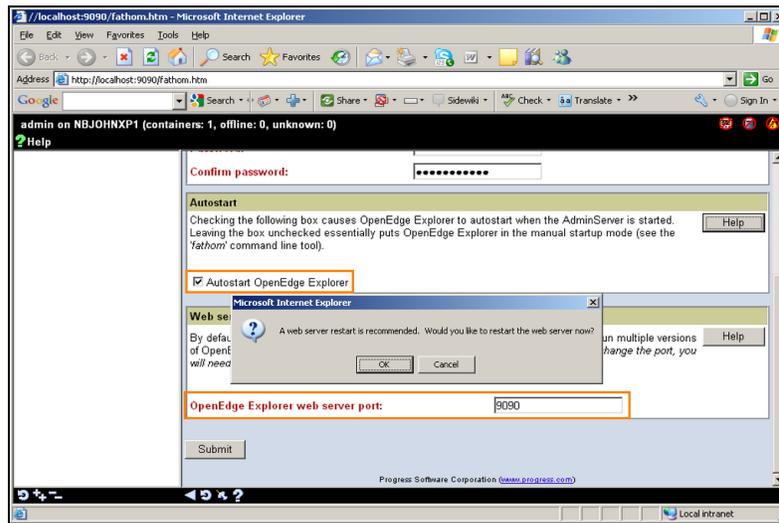
This paper accompanies the first in a series of presentations that focus on different aspects of extending an OpenEdge application to take advantage of today's tools for building what are called Rich Internet Applications, RIA for short. The series introduces the major building blocks and some of the choices you can make on the OpenEdge side, and then introduces you to some representative examples of RIA toolkits that you can connect to your ABL application to create a variety of rich UIs for different deployment situations.

This first session shows a few of the highlights of the OpenEdge Explorer tool that lets you configure AppServers and WebSpeed servers, define Web Service Adapters and deploy ABL services to them, and much more. It is similar in functionality to the Progress Explorer tool that has been in several product releases, and combines its features with OpenEdge Management. I use OpenEdge Explorer to configure a simple AppServer.

The first thing to know about OpenEdge Explorer is that when you first run the tool after installing OpenEdge, the default username and password combination to use to get in is admin / admin. And on your first entry into Explorer, you're prompted to reset the admin password, and to define whether to auto-start Explorer when the AdminServer is started, which saves you from having to do this manually, as shown here:
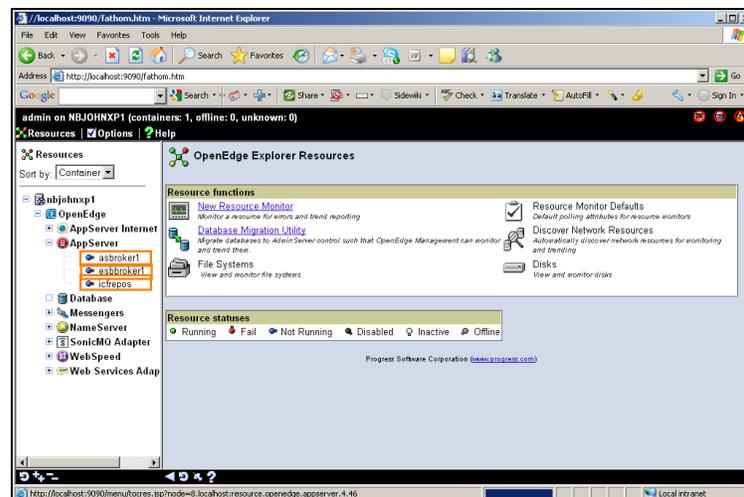
You can also confirm whether to start the Explorer's web server on port 9090. As the note on the page indicates, you might want to change this if you are running multiple versions of Explorer, but otherwise you can leave this setting at its default value. Then you click **Submit** to establish the new admin password and these two other settings. You'll see an alert box to confirm that it's alright to restart Explorer's web server, which you should OK.
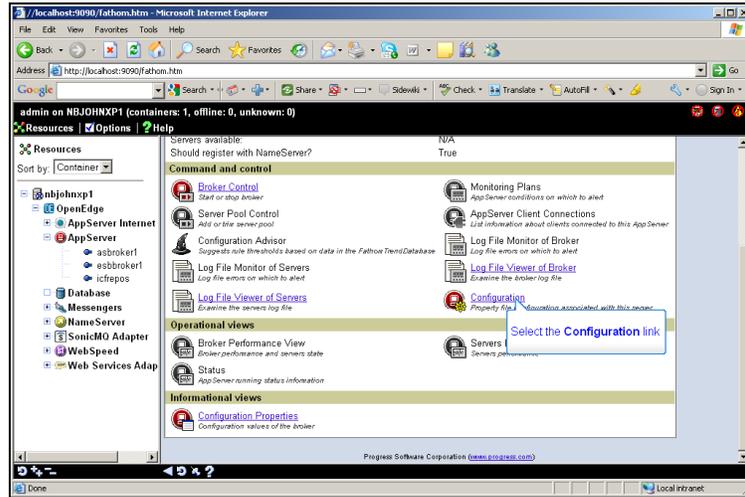

Sometimes there can be a timing error in restarting the server and then reinitializing Explorer, and you may see a "Page cannot be displayed" error in the browser. In that case, you just need to exit Explorer and restart it yourself, enter your new password, and you'll get in successfully.
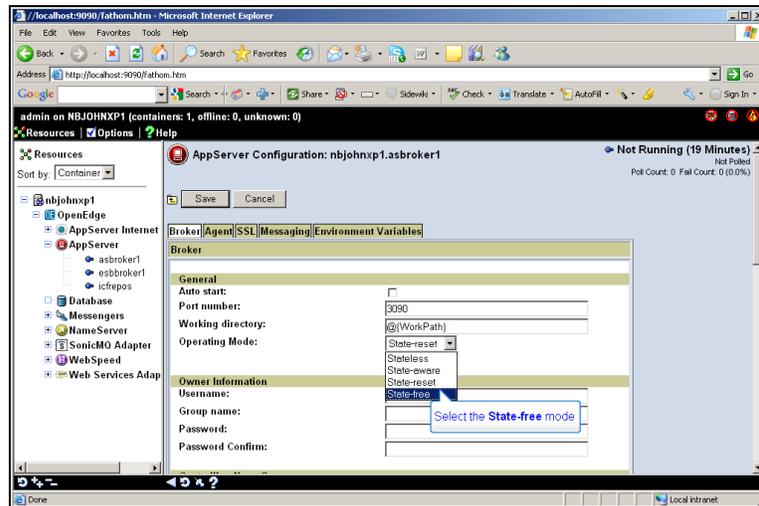
If you've used Progress Explorer to configure OpenEdge resources, most of what OpenEdge Explorer does will not look unfamiliar, but there are some differences. You can drill down into the resources on your machine, and review the steps in enabling the default AppServer broker. Here you see the familiar default AppServer for testing distributed OpenEdge applications, `asbroker1`, along with the one for providing access to OpenEdge procedures from the Sonic ESB, and the ICF repository broker for Progress Dynamics:
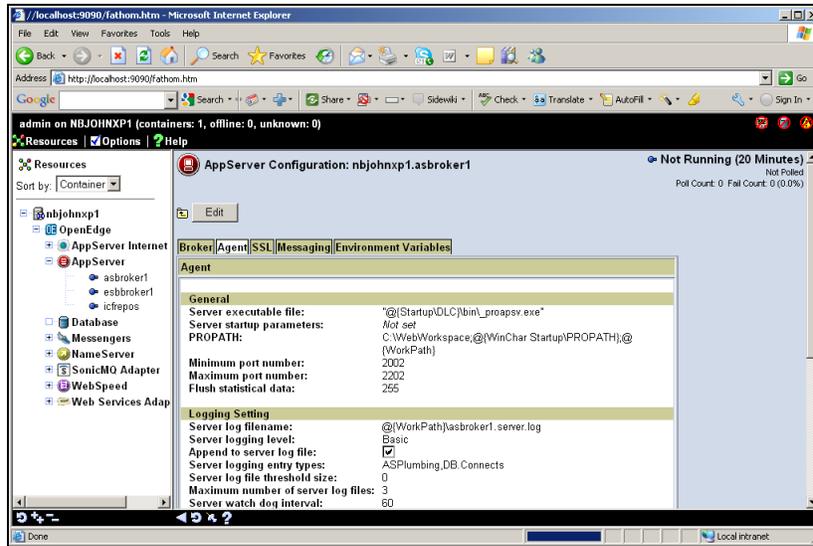
If you select `asbroker1` in the **Resources** view, the pane on the right shows its status and links to all the settings you can change. These links combine both Explorer features and OpenEdge Management operations, and this presentation just reviews a couple of the basic Explorer settings. Scrolling down so you can see all the options in the **Command and Control** section, you can start by clicking the **Configuration** link.
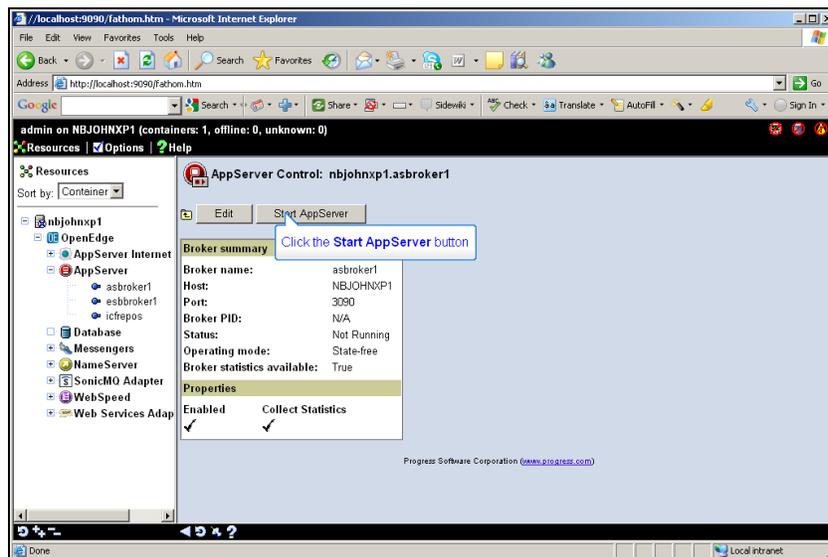
Under the different tabs are many options you can configure. We'll look at the principal ones for the Broker and then its Agents. Clicking the **Edit** button, you can first change the default **Operating Mode** from `state-reset`, which would bind an AppServer session to a connecting client, and instead select the `state-free` mode, which makes every interaction between client and server independent, a prerequisite for any sort of browser-based communication with OpenEdge.
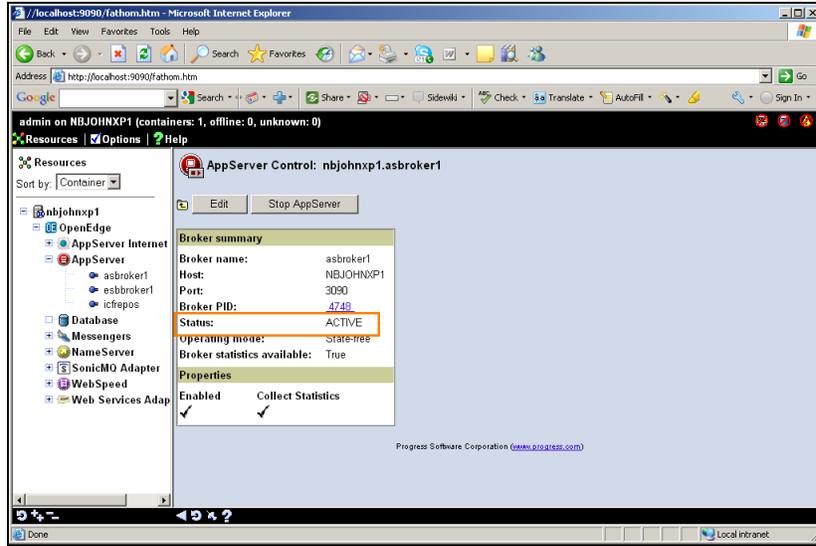
Under the **Agent** tab there are many settings that apply to the AppServer agents. Here I just make a single change, to add the parent directory of some AppServer test procedure code that I create in a later session to the head of the AppServer's ProPath, so that it can be found when it's run on the AppServer. So I add that workspace folder to the ProPath, and save that change.

Those two changes plus the defaults are all I need to enable me to test running procedures on an AppServer from a separate OpenEdge client. The one thing remaining to do is to start the AppServer broker, so it's ready to start receiving requests to run procedures on its agents. Selecting the **Broker Control** link, I just need to click the **Start AppServer** button.

Now if you've used Progress Explorer, you know that at this point you have to select the status display and keep refreshing it until it tells you that the agents are active. This **AppServer Control** display should show automatically when the Status changes to `Active`. But depending on the browser type, this may not always happen, so if it doesn't change within a few seconds, select asbroker1 again in the **Resources**, and that will show you that the broker is now active. And selecting **Broker Control** again, The ACTIVE status will display there as well.

Now your AppServer is ready for testing.

In additional sessions in this series, I show you how to do testing on an AppServer from OpenEdge Architect, how to configure a Web Services adapter and deploy ABL procedures to it as services, how to access those services from various RIA toolkits, and how to do the equivalent using WebSpeed. So there's a lot to come.