USING APPSERVER SUPPORT IN OPENEDGE ARCHITECT

John Sadd Fellow and OpenEdge Evangelist Document Version 1.0 August 2010





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 a two-part video that continues the AppServer thread in the series on building the foundation for a Rich Internet Application connection to your server-side OpenEdge business logic. The previous session on this topic – **Starting OpenEdge and Configuring a Sample AppServer** -- showed how to get into OpenEdge Explorer and use it to configure and enable an AppServer broker. This paper and its videos show you how to use the support for AppServer in OpenEdge Architect to develop and test distributed applications.

I start with a project named **webSamples** that is used in several presentations in this series. The code below is a very simple test procedure that I want to be able to run on an AppServer:

```
/* TestCust.p:
    Test procedure for AppServer and WebServices call. 10/07 */
DEFINE INPUT PARAMETER pCustNum AS INTEGER NO-UNDO.
DEFINE OUTPUT PARAMETER pCustName AS CHARACTER NO-UNDO.
DEFINE NEW GLOBAL SHARED VARIABLE giCallCount AS INTEGER NO-UNDO.
giCallCount = giCallCount + 1.
IF pCustNum = 1 THEN
DO:
    pCustName = "Test Name".
    RETURN "Success!".
END.
ELSE DO:
    pCustName = "No Name".
    RETURN "Failure!".
END.
```

It returns a name as a character string, and a success or failure return value depending on the integer value passed in. Importantly, it also defines a **GLOBAL SHARED VARIABLE** that it increments, and you'll see in a moment why this is revealing.

Next is a procedure that runs testcust.p and displays a message with the values that come back:

```
/* RunTestCust_Local.p:
    Test procedure to run TestCust without the AppServer. */
DEFINE VARIABLE cCustName AS CHARACTER NO-UNDO.
DEFINE VARIABLE hServer AS HANDLE NO-UNDO.
DEFINE VARIABLE lReturn AS LOGICAL NO-UNDO.
DEFINE NEW GLOBAL SHARED VARIABLE giCallCount AS INTEGER NO-UNDO.
giCallCount = 1.
hServer = SESSION.
RUN \WebWorkspace\WebSamples\TestCust.p ON hServer (INPUT 1, OUTPUT cCustName).
MESSAGE "Status: " RETURN-VALUE SKIP
    "Customer is " cCustName SKIP
    "Call count is " giCallCount VIEW-AS ALERT-BOX.
```

Setting the server handle to **SESSION** means that the **RUN** statement will execute locally. Notice that it also defines the same global variable and initializes it to 1. I first run this locally in the project's AVM:

| OpenEdge Editor - WebSamples/RunTest | ust_Local.p - OpenEdge Architect - C:\WebWorkspace | _ 🗆 🗙 |
|--|---|----------------|
| File Edit Source Navigate Search Project | OpenEdge Run Window Help | |
|] 🗈 • 🗄 👜 🔏 💷 🗞 • 📀 | • Q ₄ •] A •] ∯ • ∰ • ⊕ ⇔ → → | |
| 🔛 🚭 OpenEdge Editor | | |
| 💞 Resources 🖾 📃 🗆 | 🖻 testcust.p 🔹 RunTestCust_Local.p 🗙 | - 0 |
| ← → 🍖 🖨 😓 🏹 | /* RunTestCust_Local.p: | <u> </u> |
| .propath | Test procedure to run TestCust without the AppServer. */ | |
| DataTypes.html | DEFINE VARIABLE CCUSTName AS CHARACTER NO-UNDO. | |
| index.html | DEFINE VARIABLE hServer AS HANDLE NO-UNDO. | |
| provsddor.css | DEFINE VARIABLE 1Return AS LOGICAL NO-UNDO. | |
| runbproxygen.bat | | |
| RunTestCust_AS.p | dicallCount = 1 | |
| P RunTestCust_Local.p | ground | |
| RunTestCustOut_WS.n | hServer = Message (Press HELP to view stack trace) | |
| estcust.p | Status: Surress | |
| 🖻 testcustout.p | RUN \WebWo Customer is Test Name erver (INPUT 1, OUTPUT cCustName). | |
| 👕 WebSpeedSamples 🚽 | ILEDSINGE "S Call count is 2 | |
| | "C OK Help ALERT-BOX. | |
| DB S DProp | | |
| (0 🔅 🗆 | | |
| 🗄 🗐 Variables | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | 🖳 Console 🔀 [🛃 Problems 🖉 Tasks 🛛 👘 🖉 🛃 💭 🛃 💭 🛃 👘 🛃 | <u>} • ° 8</u> |
|] T* E | Writable Insert 1:1 | |
| | | |

The called procedure testcust.p returned the "Success!" return value, got a name back, and the global variable gicallcount is equal to 2. The calling procedure set it to 1, and the called procedure incremented it, as illustrated here:

| RunTestCust_Local.p |
|--|
| giCallCount = 1. |
| RUN testcust.p (INPUT 1 , OUTPUT cCustname |
| MESSAGE giCallCount = 2! |
| |
| testcust. |
| INPUT iCustNum |
| OUTPUT cCustName |
| giCallCount = giCallCount + 1. |
| |

This value just confirms that these two procedures are running in the same OpenEdge session. If the intention is to deploy this as a distributed application, with testcust.p running on an AppServer with the rest of the business logic, then it's important to integrate the AppServer into the development process and initial testing. The global variable is a very simple example of the kinds of behavior differences you can get when an application runs in multiple OpenEdge sessions. Across an AppServer connection, there will be two separate instances of gicallcount, so the result returned should not be equal to 2. More realistically, if there are AppServer procedures that pass a temp-table or a ProDataSet BY-REFERENCE, or if the application runs persistent procedures and expects to be able to access them through their handles, then the behavior can be very different when the application is actually run with the AppServer. This is why we provide support in Architect for using the AppServer as part of the development process.

Here is how you enable AppServer support for a project in Architect. Under the **Window** menu, you can open the **AppServer** perspective. What you see when you do this is two new tabs for additional Views, by default down in this pane on the bottom right. The first is the **Servers View**.

| DE OpenEdge AppServer - WebSamples/RunTestCust_Local | al.p - OpenEdge Architect - C:\WebWorkspace | | | |
|---|---|--|--|--|
| File Edit Source Navigate Search Project OpenEdge Run | n Window Help | | | |
|] 🛅 • 🗒 🖆] 🔏 💷 🦦] 🕸 • 🔕 • 🎭 •] 🛷 | 8 •] ≝ → 8 → 10 40 + → → | | | |
| 📑 🔚 OpenEdge AppServer 👫 OpenEdge Editor | | | | |
| 🗳 Resources 🛛 👘 🗖 🖻 testcust.p | RunTestCust_Local.p 🛛 | | | |
| Control (Control (Contro) (Control (Control (Control (Contro) (Control (Contro) (Contro) | INTESTCUEL LOCAL.p: Ist procedure to run TestCust without the AppSer Ist procedure to run TestCust without the AppSer IE VARIABLE MSERVENT AS HANDLE NO-WNDO. IE VARIABLE MSERVENT AS HANDLE NO-WNDO. IE VARIABLE MSERVENT AS HANDLE AND NO-WNDO. ICOUNT = 1. er = SISSICN. WebWorkspace\WebSamples\TestCust.p ON hServer "Customer is " cCustName SKIP "Customer is " cCustName SKIP "Call count is " giCallCount VIEW-AS ALERT-I Sate Sature MSERVENT | TEGER NO-UNDO. (INPUT 1, OUTPUT cCustName). BOX. | | |
| | | | | |
| | | | | |

This shows you connections that have been defined in Architect to servers such as the OpenEdge AppServer. None are displayed initially, even though there are several AppServers that are defined as part of the Openedge install. That's because you need to define connections to them from Architect. There's also a **Server Monitor View**, which shows details for any AppServers that the session is connected to. Even though it's not displaying anything useful yet, you can access OpenEdge Explorer directly from this View. If you log in to Explorer, you see it come up in the editor pane. And you can drill down into the list of AppServers to see the basic configuration information on, for instance, asbroker1.

| DE OpenEdge AppServer - http://localhost:90 File Edit Navigate Search Project OpenEdge | 90/fathom.htm - OpenEdge Architec Run Window Help | t - C:\WebWorkspace | - | |
|--|---|--|--|--------------------|
| P DenEdge AppServer | a. . Mari Vari Nari Arian Aria | | | |
| 🖓 Resources 🛛 🗖 🗖 | testcust.p RunTestCust_Lo | cal.p 😔 OpenEdge Explorer 🛛 | | |
| | admin on NBJOHNXP1 (contai | ners: 1, offline: 0, unknown: 0) | 88 | 8 |
| .propath .propath DataTypes.html index.html OperationIndex.html | Resources Voptions A | elp AppServer: nbjohnxp1.asbroker | r1 • Not Running (26 Minute Not Pol Poll Count: 0 Fail Count: 0 (0.0 | s) ▲ led 1%) |
| provsddoc.css runbproxygen.bat P RunTestCust_AS.p P RunTestCust_Local.p | □ 🔂 nbjohnxp1 □ 🕕 OpenEdge | Delete | | |
| P RunfestCustQwS.p PartestCustQut_WS.p Petcustp testcustup testcustup testcustup Testcustput Testcustput | App Server Intern App Server Stroker esbbroker esbbroker eifepos Database Messengers | App Server Status Host: Broker: Operating mode: Broker statistics available: Servers available: Should register with NameServer? | NBJOHNXP1 Not Running State-free True N/A True | |
| | Constant of the server Constant of the server Constant of the server Constant of the server Constant of the server of the | Command and control Broker Control Start or atop broker | Monitoring Plans AppServer conditions on which to alert | |
| | Console Problems 2 Tasks | Servers 🙀 OpenEdge Server Monitor 🛛 mode: Broker status: | 🔗 🥸 🔗 Add Trim Trim Al 🔞 🌣 Broker port: Broker PID: | |
| | Active agents: Busy agents: Locked agents: Agent Summary Agent Detail | Active clients: Client queue: Total requests: | | |
|) 1° E | | | | |

If you go back into **Configuration** information, there are settings for the broker, including its working directory, and information for the agents that run in that broker, including their ProPath.



This shows you that Architect provides you with access to a lot of useful information before you even configure anything for a specific project. But what this paper describes is how to provide AppServer support as part of a project, such as the **websamples** project, so that you can run procedures such as **testcust.p** in an AppServer while doing development. To do that, you need to associate an AppServer with a project, and tell Architect which code in the project is going to be deployed to the AppServer.

This requirement results in a bit of a chicken-or-the-egg problem. You can't name an AppServer connection as the one to use for a project, because there aren't any server connections until you define them. But if you define a connection, you can't yet name the Architect projects that should use it, because you haven't told architect that they should be AppServer-enabled. So you need to start at one end of the process or the other and then go back and fill in the missing pieces.

Start by defining a new Server in the **Servers View**. Right-click in the View, and select the only option: **New -> Server**.

| DE OpenEdge AppServer - WebSamples/Run | 'estCust_Local.p - OpenEdge Architect - E:\WebWorkspace | _ 🗆 × |
|--|---|--------|
| File Edit Source Navigate Search Project | OpenEdge Run Window Help | |
| 🗈 • 🗄 🗎 🔟 🖾 🕲 🏇 • 🕗 | • Q •] A •] 拉 · 拉 · 拉 · · · · · | |
| 🗈 🔚 OpenEdge AppServer 😽 OpenEdge Edit | or | |
| 💞 Resources 🔀 🗖 🗖 | 🖻 testcust.p 🛛 🖗 RunTestCust_Local.p 🔀 | - 8 |
| ← → ∞ 📄 😘 🏹 | / * RunTestCust_Local.p: | ~ |
| .propath | Test procedure to run TestCust without the AppServer. */ | |
| DataTypes.html | | |
| 📔 index.html | DEFINE VARIABLE CCustName AS CHARACTER NO-UNDO. | |
| 🗈 OperationIndex.html | DEFINE VARIABLE hServer AS HANDLE NO-UNDO. | |
| - 🖹 prowsdidoc.css | DEFINE VARIABLE IReturn AS LOGICAL NO-UNDO. | |
| - 💽 runbproxygen.bat | DEFINE NEW CLOBAL CHARGE MARTARIE FIGHTIGHTER AC INTEGER NO. INDO | |
| P RunTestCust_AS.p | DEFINE NEW GODAL SHARED VARIABLE GIGHTCOURC AS INTEGER NO-ONDO. | |
| P RunTestCust_Local.p | gitalicount - 1. | |
| RunTestCust_WS.p | beautres = STSSTON | |
| RunTestCustOut_WS.p | IDELVEL - SESSION. | |
| - 🖻 testcust.p | DIN \ WebUorkenace) WebSamples) Test Cust n (N bSarver (INDUT 1 OUTDUT COustName) | |
| P testcustout.p | WFSAGE "Status," ptillestill stip | |
| | "Customer is " cCustName SKID | |
| 🗄 Outline 🛛 📑 DB S 🗔 Prop 🖓 🗖 | "Call count is " giCallCount VIEW-AS ALERT-BOX. | |
| — 《 本 ② | | |
| 🗄 🔞 Variables | | - |
| | 4 | |
| | 🖳 Console 🔝 Problems 🖉 Tasks 🕷 Servers 🔀 🎇 OpenEdge Server Monitor 👘 🕠 🜮 😑 | 10 - 0 |
| | Server A State Status | |
| | | |
| | | |
| | | |
| | New Free Server | |
| | | |
| | | |
| | | |
| | | |
| | | |

The use of the word *server* here is somewhat confusing. You can't in fact define a new AppServer in Architect, though you can access and even modify a lot of its properties. What's called a server here is really a *connection* from Architect to an existing AppServer. Just keep that in mind.

The **New Server** wizard comes up:

| OF OpenEdge AppServer - WebSamples/RunT | estCust_Local.p - OpenEdge Architect - C:\WebWorkspa | ace | |
|---|---|-------------------------------------|-----------------------|
| | | | |
| | 0E New Server | - [0] × | [|
| | Define a New Server | | - 8 |
| | Choose the type of server to create | _ | |
| DataTypes.html | Server's host name: ⁰ localhost | | (|
| OperationIndex.html | Select the server type: | Download additional server adapters | |
| runbproxygen.bat | type filter text | | O-INDO. |
| P RunTestCust_AS.p P RunTestCust_Local.p P RunTestCust_WS.p P RunTestCustOut_WS.p | Besic Progress Software Corporation A OpenEdge AppServer | | |
| P testcust.p P testcustout.p WebSpeedSamples | | | 1, OUTPUT cCustName). |
| 6 Outline X 10 DB 5 □ Prop □ U | Publishes ABL code to an OpenEdge application server | | |
| 🗉 🕥 Variables | | | |
| | Server name: OpenEdge 10.2B at localhost | | |
| | Server runtime environment: OpenEdge 10.28 | ▼ Add | |
| | | Configure runtime environments | |
| | | | |
| | (?) <br< td=""><td>Finish Cancel</td><td></td></br<> | Finish Cancel | |
| | | | |
|] ➡ 🖻 | | | |

On the first page of the **New Server** wizard, the only host name you can use is **localhost**. Architect doesn't yet support publishing code to an AppServer on another machine. Keep in mind that Architect is primarily a development tool, not a deployment management or performance testing tool, and **localhost** is all you need to test the critical elements of application logic that depend on operations being run across the AppServer boundary.

The **server type** filter is set up by default to let you define an AppServer – that's the only kind of server connection supported in OpenEdge 10.2B, at least. Provide the server with an appropriate name. Since this is a connection to **asbroker1**, this example uses the name **AsBroker1conn**. Continue on to the next page in the wizard.

| | | 1=1-4 |
|---|--|-------|
| Ele Edit Source Navinate Search Project (| estLust_Local.p - OpenEdge Architect - L:\webworkspace | |
| | • Q. • A • 5 · 5 · 6 • • • | |
| ConenEdge AnnServer | 0E New Server | |
| Resources X | Define a new AppServer broker | |
| Creath Control Control Contro Contro Control Control Control Control Con | OpenEight a deale the relate of the hypothetic terms Seven main: [AStroker Come OpenEight AdmitSeven comedian Hotz: [Penhome] Pot multiple Passond: Broker: schrider1 Broker: schrider1 Comedian | |
|] t* 🗈 | | |

Next press the **Select...** button to locate the AppServer broker. This brings up a dialog to display the brokers on your machine. For the example I select **asbroker1**. The **Port number** is the port that the AdminServer runs on, not the AppServer. That should be initialized correctly, so it can just be left at this value. Continue on to the next page of the wizard.

| OE OpenEdge AppServer - WebSamples/RunTo File Edit Source Navigate Search Project C | stCust_Localp - OpenEdge Architect - C:\WebWorkspace perEdge Run Window Heb | -0× |
|---|--|-----|
| OpenEdge AppServer Structure Structure | Variation Variation Øg New Server Image: A server Define publishing rules Image: A server | |
| Resources SS propath propath propath provediac csa provediacsa provediac csa provediacs | Deture publishing rules A server publish directory must be provided. Publish biotecomposition of textory Publish directory Publish directory Publish directory Publish directory Publish directory Publish directory Publish directory Publish directory Publish directory Publish directory Publish directory Publish directory Publish directory Publish directory Publish directory Publish directory Publish r-code I, OUTPUT occust Name). | 1 |
| ♥ Variables □ ° (%) | (Stack Next > First) Cancel | |

Here you tell Architect where to publish code. This is one of the elements that's incomplete until you define both the server connection on the one hand and the properties for an AppServer project on the other. What this setting means is: for any project that you associate with this AppServer connection, that project is going to be prepared to publish -- to copy -- ABL procedures or classes from a designated set of directories in the project to a target set of directories that the AppServer will see. This is part of the separation of the project's client OpenEdge session and the AppServer session. For my example I could select the AppServer's working directory as the target, which is **openEdge102B\WRK**. But that's not where I want the code to go. I've created a folder on my machine where I want published code to go, so I enter its name here: C: **AppServerDeploy**. I need to make sure this is in the ProPath of the AppServer, which is a setting I can make a little later.

There's another important point to make here which this diagram illustrates:



The *publish directory* is where code gets copied to after you create or make changes to it. It's not where you make those changes. So you shouldn't put this folder somewhere in the project or the workspace the project is in. In fact, you'll get an error if you try to do that. It needs to be independent of the project itself.

Finally, in the wizard you can choose whether you want both a copy of the source code and the compiled r-code to be published.

Continue on to the last page of the New Server wizard.

| 0E OpenEdge AppServer - WebSamples/RunT | estCust_Local.p - OpenEdge Architect - C:\WebWorks | | _ _ × |
|--|--|----------------------|-----------------------|
| File Edit Source Navigate Search Project O | OpenEdge Run Window Help | | |
|] 📑 • 🗟 📥 🛛 🔏 📾 🌚] 🏇 • 🕑 | • Q₄ • A? • 21 × 21 × 50 (0 • + → × - | | |
| 🗈 🔚 OpenEdge AppServer 💐 OpenEdge Edito | OE New Server | <u>_ 0 ×</u> | |
| 🥰 Resources 🛛 🗖 | Add and Remove Projects | | - 8 |
| | Modify the projects that are configured on the server | | × |
| | | | / |
| DataTypes.html | Move projects to the right to configure them on the server | | |
| index.html | Available projects: | Configured projects: | |
| OperationIndex.html prowadding.com | | | |
| Tunbproxygen.bat | Add > | 1 | |
| RunTestCust_AS.p | | 1 | o-undo. |
| P RunTestCust_Local.p | Remove | 1 | |
| RunTestCust_W5.p | | J | |
| estcust.p | | | |
| testcustout.p | | | 1, OUTPUT cCustName). |
| | Add All >> | | |
| 🗄 Outline 🐹 🖬 DB S 🔲 Prop 🖓 🗖 | | | |
| () () | << Remove Al | | |
| Variables | | | _ |
| - | | | × |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | (?) <back :<="" next="" th=""><th>Finish Cancel</th><th></th></back> | Finish Cancel | |
| | | | u |
| L | L | | |
|) T* 🖻 | | | |
| L | | | |

Here's where you see how the process of defining the first connection isn't complete until you go into the project properties as well. There are no available projects to be associated with this server connection, because you have to turn on AppServer support in one or more projects for them to be eligible to show up on this list. So just leave this alone for the moment, and press **Finish** to complete the definition of the new server connection. Below you can see the new server connection in the **Servers** view.

| 0E OpenEdge AppServer - WebSamples/Ru | nTestCust_Local.p - OpenEdge Architect - C:\WebWorkspace | _ 🗆 × |
|--|---|--------------|
| File Edit Source Navigate Search Project | OpenEdge Run Window Help | |
| 📑 • 🖫 🛎 🛛 🗶 📖 🗞 • 🖸 |) • Q ₄ • _A / • _A → A → A → A → A → A → A → A → A → A → | |
| CoenEdge ApoServer | litor | |
| | 1 Distantes Distanticut Leaden M | |
| Resolutes 23 | rescarp Francescos otarp a | - |
| · · · · · · · · · · · · · · · · · · · | /* Runiestcust_Local.p: | <u> </u> |
| - i propath | Test procedure to run rescust without the apperver7 | |
| DataTypes.html | DEFINE VARIABLE COustName AS CHARACTER NO-UNDO. | |
| Index.ntml OperationTendex.html | DEFINE VARIABLE hServer AS HANDLE NO-UNDO. | |
| operationundex.nom | DEFINE VARIABLE IReturn AS LOGICAL NO-UNDO. | |
| I runhorovygen bat | | |
| RunTestCust AS.p | DEFINE NEW GLOBAL SHARED VARIABLE giCallCount AS INTEGER NO-UNDO. | |
| RunTestCust_Local.p | giCallCount = 1. | |
| RunTestCust_WS.p | | |
| RunTestCustOut_WS.p | hServer = SESSION. | |
| P testcust.p | DIN \ HebNorkenege\ HebSorp \est Test Cust n ON bServer (INDUT 1 OUTDUT aCust Nere) | |
| P testcustout.p | Now (desworkspace) descapace (descapace) is the set of | |
| WebSpeedSamples | "Customer is " cCustName SKIP | |
| 🗄 Outline 🛛 📅 DB S 🖾 Prop 🖵 🛙 | "Call count is " giCallCount VIEW-AS ALERT-BOX. | |
| (0 🌼 🗉 | | |
| 🗉 🔞 Variables | | - |
| | T | |
| | 🕒 Console 🔝 Problems 🖉 Tasks 🤻 Servers 🔀 👯 OpenEdge Server Monitor 🛛 🕸 💽 🖉 | <u>ا ۳ و</u> |
| | Server State Status | |
| | 🧏 ASBroker1Conn 🖁 Stopped Republish | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| T [♦] E | | |
| | | |

To complete the connection you must make the needed changes to one of your projects. For this example, the first thing I want to do is define a new folder in the project where code to be published to the AppServer should go. This is where I will develop those procedures or classes, so they are part of the project. Right-click on the project, and create a new folder, named for example AppServerContent. For starters, I want to move my little procedure testcust.p to that new folder, so that it will get published to the AppServer folder. So here it is in the new AppServerContent folder.

| OpenEdge AppServer - WebSamples/RunT | FestCust_Local.p - OpenEdge Architect - C:\WebWorkspace | | | | |
|---|--|------------|--|--|--|
| File Edit Source Navigate Search Project OpenEdge Run Window Help | | | | | |
|] 💼 • 🛛 △] 🔏 म 🕲 🎄 • 🕖 • 💁 • 🖉 • 1 ₂ • 1 = - 1 = | | | | | |
| 🗈 🚍 OpenEdge AppServer 🖑 OpenEdge Edito | ar | | | | |
| 🖏 Resources 🛛 🖓 🗖 | 🕑 testcust.p 🛛 🖗 RunTestCust_Local.p 🕴 | | | | |
| $(- \rightarrow \otimes) = {}^{\bullet} \nabla$ | /* RunTestCust_Local.p: | ^ | | | |
| 🖻 👹 WebSamples 🔺 | Test procedure to run TestCust without the AppServer. */ | | | | |
| Procedure Libraries | DEFINE VARIABLE COURTName AS CHARACTER NO-INDO. | | | | |
| 🗄 🗁 .settings | DEFINE VARIABLE DEFINET AS HANDLE NO-INDO. | | | | |
| E AppServerContent | DEFINE VARIABLE IReturn AS LOGICAL NO-UNDO. | | | | |
| | DEFINE NEW GLOBAL SHARED VARIABLE giCallCount AS INTEGER NO-UNDO. | | | | |
| | giCallCount = 1. | | | | |
| DataTypes.html 🛁 | 10 07007/M | | | | |
| 🔝 index.html | haelvel - alasion. | | | | |
| | RIN \ MebNorkspace \ MebSamples \ TestCust. p. ON bServer (INPUT 1. OUTPUT cCustName). | | | | |
| prowsdidoc.css | MESSAGE "Status: " RETURN-VALUE SKIP | | | | |
| runoproxygen.bat | "Customer is " cCustName SKIP | | | | |
| 🗄 Outline 🖄 📑 DB S 🗔 Prop 🖓 🗖 | "Call count is " giCallCount VIEV-AS ALERT-BOX. | | | | |
| (i) 🍄 😑 | | | | | |
| 🗉 🔇 Variables | | - 1 | | | |
| | <u>(</u> | | | | |
| | 🕒 Console 🔝 Problems 🖉 Tasks 👭 Servers 🖾 🎬 OpenEdge Server Monitor 🛛 🕸 🔊 😑 💬 | - 8 | | | |
| | Server - State Status | | | | |
| | ASBroker1Conn a Stopped Synchronized | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| 1 * € | n | | | | |
| | | | | | |

Once again, this is the source directory for publishing code. The AppServerDeploy folder that I defined as part of the server connection is the target directory. Next you need to take the step I've been referring to, enabling AppServer support for this project. Right-clicking on the project, select **OpenEdge**, and **Add AppServer support**.

| E OpenEdge AppServer - WebSamples/RunTes | ust_Local.p - OpenEdge Architect - C:\WebWorkspa | te | | _02 |
|--|--|------------------------|----------------|-------------|
| File Edit Source Navigate Search Project Ope | Edge Run Window Help | | | |
| 📑 • 🗄 📄 🛛 🔏 📖 🗞 🖉 • 🕥 • ' | ⊾ •] | | | |
| 🗈 🚍 OpenEdge AppServer 💐 OpenEdge Editor | | | | |
| 🖏 Resources 🕱 👘 🗖 | testcust.p 🖻 RunTestCust_Local.p 😒 | | | |
| | /* RunTestCust Local.p: | | | * |
| E-12 WebSam | Test procedure to run TestCust w | ithout the AppSe | rver. */ | |
| E-JL Proce | | | | |
| 🕀 🗁 .settir 🛛 🕞 Go Into | DEFINE VARIABLE COustName AS CHARAC | TER NO-UNDO. | | |
| Den in New Window | DEFINE VARIABLE IServer as HANDLE | NO-UNDO. | | |
| E Copy | DEFINE VARIABLE INCOLL AS BOOICA | L NO-0NDO. | | |
| Paste | DEFINE NEW GLOBAL SHARED VARIABLE g | iCallCount AS IN | FEGER NO-UNDO. | |
| Delete | giCallCount = 1. | | | |
| DataT Move | | | | |
| index Rename | hServer = SESSION. | | | |
| - Opera Ne Import | | | | |
| prows A Export | MESSAGE "Status: " RETURN-VALUE SKIP | | | |
| runbp | "Customer is " cCustName SK | IP | | |
| 🗄 Outline 🔀 🕴 Refresh | "Call count is " giCallCoun | t VIEW-AS ALERT- | BOX. | |
| Close Project | | | | |
| Close Unrelated Projects | | | | |
| Validate | | | | <u></u> |
| Team 🕨 | | | | <u> </u> |
| Compare With | Console 🚺 Problems 🖉 Tasks 端 Servers 🛛 🎇 G | penEdge Server Monitor | | 🎋 🜔 🖉 🗏 🔛 🗖 |
| Restore from Local History | rver ÷ Qtatp | Status | | |
| OpenEdge | Add AppServer support topped | Synchro | nized | |
| Make Catalog | Add Annotation | | | |
| Web Development Tools | Restart OpenEdge AVM | | | |
| Source | Compile | | | |
| | Clear OpenEdge Compile Errors | | | |
| Properties | Generate Sonic ESB Invocation Files | | | |
| | | | | |
| T WebSamples | | | | |

In the example, WebSamples is the project. Select the parent source folder to publish content from. This is all the code that will be published to the AppServer for testing as it's developed. In the example that's the AppServerContent folder I created in the project. Now that you've defined an AppServer connection for Architect, you can select it as the one to use for your project.

| DE OpenEdge AppServer - WebSamples/RunTestCust_Local.p - OpenEdge Architect - C:\WebWorkspace | _0× |
|---|--------------------------|
| File Edit Source Navigate Search Project OpenEdge Run Window Help | |
|] <mark>□ + □ △] ∡ □ </mark> | |
| 📋 📄 OpenEdge AppServer 🖏 OpenEdge Editor 🛛 🚺 Add AppServer support to OpenEdge project. | 1 |
| Select AppServers | |
| Select the servers to which the project's AppServer module is unbished. | A |
| P 🔁 WebSamples | |
| B-JL Procedure Libraries Supported servers: | |
| Server name | |
| Estcust.p | |
| 📄 .dbconnection | IN NO INDO |
| | R NO-ONDO. |
| | |
| - Diada yees.nom | |
| | |
| 🖻 prowsdidoc.css | UT 1, OUTPUT cCustName). |
| Publish changes inmediately | |
| 🗄 Outine 🕮 📅 DB S 💷 Prop 🙄 🗖 | |
| | |
| H 🔇 Variables | |
| Cancel Cancel Cancel | <u> </u> |
| Console 👔 Problems 🗹 Tasks 🕫 Servers 23 See OpenEdge Server Monitor | |
| Server A State Status | |
| 🕌 ASBroker1Conn 🔓 Stopped Synchroniz | ed |
| | |
| | |
| | |
| | |
| | |
| | |
| U Le WebSamples | 1 |

You can ask it to publish any content that is already there, and you're done. That's all there is to it. Now this project is AppServer aware. Architect will automatically publish procedures, classes, and their r-code to a separate directory in the AppServer's propath, so that you can test client-side ABL running those procedures on the AppServer.

Now I will explain how to make changes to these new properties. In the project properties, expand the **OpenEdge** node, and then expand the node labelled **Modules**. A module is a set of related files organized under a directory, in this case all the files that you want to publish to the AppServer. For example, if I select the WebSamples module, I can then modify the pointer to the top-level folder that holds all the AppServer code.

| De OpenEdge AppServer - WebSam Fie Edt Source Navigate Search 📸 - 📰 🚔 🔏 📖 🗞 | iples/RunTestCust_Local.p - Oper Project OpenEdge Run Windov 称 • O • Q • J ∻ • J ∲ | iEdge Architect - E:\WebWorkspace / Help → 褶 → 🏷 🗘 + 🔿 → | | × |
|---|---|---|--------------|---------------|
| 😭 🔚 OpenEdge AppServer 🗳 Op | OE Properties for WebSamples | | _ _ X | |
| 💱 Resources 🕱 | type filter text | Modules | (⇒ • ⇒ • ₹ | - 0 |
| Country of the second sec | Passura Padders | □ \\[\frac{1}{2}\] Oper-Educ ArgoServer modules □ \[\frac{1}{2}\] \[\frac{1}{2}\] \[\frac{1}{2}\] □ \[\frac{1}{2}\] \[\frac{1}{2}\] \[\frac{1}{2}\] □ \[\frac{1}{2}\] \[\frac{1}{2}\] \[\frac{1}{2}\] □ \[\frac{1}{2}\] \[\frac{1}{2}\] \[\frac{1}{2}\] □ \[\frac{1}{2}\] \[\frac{1}{2}\] \[\frac{1}{2}\] | Modfy | T cCustName). |
| | | | | |
| | 0 | | OK Cancel | |
| | | | | |
| C WebSamples | | | | |

You can also select **Project Facets**. A facet is in effect a marker that is used to identify a project as being of a certain type.

| OF OpenEdge AppServer - WebS File Edit Source Navigate Sea | amples/RunTestCust_Local.p - OpenEdge Architect - C:\WebWorkspace rch Project OpenEdge Run Window Help | × |
|--|---|--|
| 🗈 • 🔛 📥 🔏 💷 🗞 |] 🌼 • 🖸 • 9₄ •] 🖉 •] 🥘 + 🦉 • 😓 🗢 • → + | |
| DE Properties for WebSamples | | |
| type filter text | Project Facets | $\diamond \bullet \bullet \bullet \bullet$ |
| Resource Builders | Configurations: <pre><custom></custom></pre> | Save As Delete |
| OpenEdge ApoBuilder | Project Facet Version | Details Runtimes |
| - Assemblies | JavaScript Toolkit 1.0 | 📄 JavaScript Toolkit 1.0 |
| Butt Butt | 10 | Adds advanced JavaSorpt eding support. |
| | | Revert Apply |
| 0 | | OK Cancel |
| | | |
| C WebSamples | | I |

Here you can see **OpenEdge AppServer** as one of several facets that define this project. Adding AppServer support to the project checked this facet on. You could remove AppServer support from the project by checking the facet off. Any OpenEdge project has the **OpenEdge** facet set, and you can't turn it off.

Now we need to complete the testing process and show some more ways to manage AppServers from within Architect. First, if you just double-click on the New Server connection, that brings up an **OverView** display with a lot of information. It confirms the information supplied when you defined the connection. If you expand the **Publishing** item, you can see and modify the options for when code changes are published, and under **Publish Location**, you can likewise confirm or modify where files get published to in order for the AppServer to find them.

| DE OpenEdge AppServer - ASBroker1Conn - File Edit Navigate Search Project OpenEr | OpenEdge Architect - C:\WebWorkspace Ige Run Window Help | |
|---|---|--|
|] 📬 • 🔛 👜] 🔏 📖 🕲] 🏇 • 🕻 |) • Q₂ •] A? •] ½ + ⅔ + % ↔ ↔ → → | |
| 🔛 🚍 OpenEdge AppServer 💐 OpenEdge E | litor | |
| 🖑 Resources 😒 🔍 🗖 E | e testcust.p P RunTestCust_Local.p ASBroker1Conn 🛛 | - C |
| ← → @ □ | Uverview | |
| WebSamples WebSamples Duration | General Information Specify the host name and other common settings. | Publishing Modify settings for publishing |
| B 🗁 .settings AppServerContent | Server name: ASBroker1Conn Host name: localhost | O Never publish automatically |
| P testoust.p .doconnection | Runtime Environment: OpenEdge 10.28 | Automatically publish when resources change Publishing interval (in seconds): 15 |
| - ipropath 100 DataTypes.html - | Connection | ▶ Timeouts |
| - index.html | Specify the information for connection to the AdminServer. | Publish Location Specify the server publish directory. |
| runbproxygen.bat | Port: 20931 | Use AppServer working directory Use custom publish directory |
| E Outline 🛛 📅 DB S 🗔 Prop 🖓 D | Password: | Publish directory: C:\AppServerDeploy |
| | Broker name: asbroker1 | Publish source code Publish r-code |
| | Overview | |
| | Eonsole 🔝 Problems 🕢 Tasks 👫 Servers 🖾 🎇 OpenEdge Se | erver Monitor 🔅 💽 🖉 🗖 |
| | Server A State | Status |
| | A Astrony Loon Co Scoped | synchronolog |
| T [*] € | л | Writable |

Importantly, over on the left is a link labeled **Open launch configuration**. Remember that in Architect, a Launch Configuration or Run Configuration holds all the options for how code is run: what the top-level procedure is to invoke, the ProPath, database connections, and so forth. So here as well, you can customize the details of how the AppServer is accessed from a project that's AppServer enabled. You can edit the ProPath the AppServer agents use. The tab shown below for instance shows ProPath for **AsBroker1conn**. I can change that to place the **AppServerDeploy** folder that I have now specified as the publish location at the head of the ProPath.

| OE OpenEdge AppSe | rver - ASBroker1Conn - OpenEdge Architect - C:\WebWorkspace | _ 🗆 🗙 |
|-------------------------|---|-------|
| File Edit Navigate | 0E Edit Configuration | 1 |
|] 📑 📲 🔚 📄] | Edit launch configuration properties | |
| 📑 🚍 OpenEdge A | Create a configuration to launch an OpenEdge AppServer. | |
| Resources 23 | | |
| | | |
| E. 7 Wah Sampler | Name: ASBroker1Conn | |
| ⊕ JL Procedu | 🗄 Server 🗱 Startup 婿 PROPATH 🤣 Databases 🗔 Common | |
| 🖲 🗁 .setting: | | |
| 😑 🗁 AppServ | the ClyApperverteptoy How Do Date | |
| dhoonn | B G (WorkPath) Move Down | |
| .project | Remove | 15 🛨 |
| .propath | Cdr. | |
| DataTyp | | |
| - Operatic | Add Procedure Library | |
| 📄 📄 prowsdk | Add External Library | 1 11 |
| Tunbpro: | Add Workspace Directory | |
| 🗄 Outline 🖾 💀 | Additional Division | |
| An outline is not avail | Add External Directory | |
| | Add Standard Paths | - |
| | Add Project | |
| | Copy ini Settings | |
| | | |
| | | |
| | | |
| | | |
| | Apply Revert | |
| | | 1 |
| | (2) OK Cancel | 1 1 |
| | | I |
| JU 19 - | writade | - |

This is one example of the AppServer properties that you can now set directly from Architect as well as from within Explorer. You can customize the database connections the AppServer uses, as well as many other options. That's all from the **OverView** you access by double-clicking on the connection in the **Servers View**.

There are some more things you can do from the **Servers View**. When you select a server connection, there are a number of buttons in this View's toolbar and in the **Server Monitor** to control the AppServer. You can start the AppServer from here, and the **Console View** will pop to the top to display the details of the startup process. One of the other buttons lets you override that behavior of the **Console View** coming up, depending on your preference.



In the **Server Monitor View**, now that there's a connection to select, all the broker details are displayed. If you select the **Agent Detail** tab, you see the state of each of the AppServer agents.

| OpenEdge AppServer - WebSamples/Run | TestCust_Local.p - OpenEdge Architect - C:\WebWorkspace | _ 🗆 🗵 |
|--|---|-------|
| File Edit Source Navigate Search Project | OpenEdge Run Window Help | |
| 🛅 • 🛛 🗠 🔏 📖 🎭 🏇 • 🔾 | • Q_ • [- A •] - 2 - 3 - 3 - 4 | |
| 📑 🚍 OpenEdge AppServer 🖉 OpenEdge Ed | tor | |
| Resources 🛛 | P testoust.p P RunTestCust Local.p 🐰 | - 0 |
| | /* RunTestCust Local.p: | - |
| E 1 ²⁴ MahSamplar | Test procedure to run TestCust without the AppServer. */ | |
| B JL Procedure Libraries | | |
| 🗄 🗁 .settings | DEFINE VARIABLE cCustName AS CHARACTER NO-UNDO. | |
| 🖻 🧁 AppServerContent | DEFINE VARIABLE ASERVEF AS RANDLE NO-UNDO. | |
| E testcust.p | berine twittee Recurn vs borton wo-wbo. | |
| .dbconnection | DEFINE NEW GLOBAL SHARED VARIABLE giCallCount AS INTEGER NO-UNDO. | |
| X .project | giCallCount = 1. | |
| DataTunar html | | |
| index.html | hServer = SESSION. | |
| | | |
| - B provisdidoc.css | RUN \WebWorkspace\WebSamples\TestCust.p ON hServer (INPUT 1, OUTPUT cCustName). | |
| 🐨 runbproxygen.bat 💌 | RESSAUE "Status: " REIORN-VALUE SKIP | |
| 🗄 Outline 🖾 📑 DB S 💷 Prop 🐃 🗆 | "Call count is " giCallCount VIEW-AS ALERT-BOX. | |
| () [*] E | | |
| 🗉 🔇 Variables | | - |
| | | F |
| | 🖸 Console 🦹 Problems 🖉 Tasks 👫 Servers 🎇 OpenEdge Server Monitor 🛛 🔗 🍄 🖑 🦉 | ~ - 8 |
| | ASBroker1Conn | |
| | Broker name: asbroker1 Operating mode: State-free Broker status: ACTIVE Broker part: 3090 Broker PID: 912 | :8 |
| | PID State Port Requests Received Sent Started Last Change | 1 |
| | 10424 AVAILABLE 2002 0 0 0 Jun 22, 2010 12:24 Jun 22, 2010 | 12:24 |
| | 🗌 💠 12128 AVAILABLE 2003 0 0 0 Jun 22, 2010 12:24 Jun 22, 2010 | 12:24 |
| | | • |
| | Agent Summary Agent Detail AppServer State | |
| 1 10 10 | | |
| 1.0.00 | · · · · · · · · · · · · · · · · · · · | |

The **Add** button in the **Server Monitor View** toolbar would let me add agents, and the **Trim** button would let me trim the number of agents running in the broker. So you can carry out almost all of the essential operations on an AppServer directly from here in Architect, in addition to being able to fire up OpenEdge Explorer from Architect as well.

To confirm the behavior of the publish operation and the source and target folders I defined earlier, I can make a simple change in my example to testcust.p and save that change, which now in addition to compiling the file will automatically publish it to the AppServerDeploy folder. I can confirm that this worked by taking a look at the file in the target folder. Below you can see the AppServerDeploy folder. Remember that I set this as the publish location for the server connection. Any AppServer-enabled project that uses this server connection will publish changes from its AppServerContent folders to this destination. The webSamples project has an AppServerContent folder that's defined as the source for publishing, so a copy of that has been created here as a subdirectory under AppServerDeploy. Here you can see the source file and r-code for testcust.p:



If I open the source file in this publish directory I can see the change that I made to change the first **RETURN** statement to return the string "Success on asbroker1!". This confirms that the files were copied over as soon as the change was saved and compiled.



Remember that when I first defined this server connection, the last screen in the wizard came up to let me associate one or more AppServer projects with the connection. And there weren't any, because I had to add AppServer support to a project before it would show up. Here you can see now that after selecting **Add and Remove Projects** for the server, the one project that I've added AppServer support to shows up in the list.



It's on the right-hand side, of course, under **Configured projects**, because I made the association to this server connection when I added AppServer support to the project. But I came back in here just to reinforce the notion that once you have defined one or more server connections, and AppServer-enabled one or more projects, then you will be able to make associations between them in the server properties or the project properties.

Now that I've made all the right connections and started the AppServer, it's time to test whether what I've done in my example works the way I expect. I open a version of my runner procedure, which represents the client side of the application. I've added a statement to connect to **asbroker1**, and a message to confirm that the initial AppServer connect succeeds.

```
RunTestCust AS.p:
   Test procedure to run TestCust. */
DEFINE VARIABLE cCustName AS CHARACTER
                                           NO-UNDO.
DEFINE VARIABLE hServer AS HANDLE
DEFINE VARIABLE lReturn AS LOGICAL
                                           NO-UNDO.
                                           NO-UNDO.
DEFINE NEW GLOBAL SHARED VARIABLE giCallCount AS INTEGER NO-UNDO.
giCallCount = 1.
CREATE SERVER hServer.
lReturn =
hServer:CONNECT ("-AppService asbroker1 -H localhost -S 5162 -sessionModel session-
free").
MESSAGE "Connecting? " lReturn VIEW-AS ALERT-BOX.
IF lReturn THEN
DO:
    RUN AppServerContent\testcust.p ON hServer (INPUT 1, OUTPUT cCustName).
    MESSAGE "Status: " RETURN-VALUE SKIP
            "Customer is " cCustName SKIP
            "Call count is " giCallCount VIEW-AS ALERT-BOX.
    hServer:DISCONNECT().
END.
DELETE OBJECT hServer.
```

When I run this procedure, the Connecting message is displayed, and then the message that shows the values that came back.

| OF OpenEdge AppServer - WebSamples/RunT | estEust_AS.p - OpenEdge Architect - C:\WebWorkspace | | | |
|---|---|-----|--|--|
| Hie Edit Source Navigate Search Project OpenLidge Run Window Hep | | | | |
|] 🖸 • 🗄 🔤] 🔏 📖 🥥] 🅸 • 🖸 | • A •] & •] 8 • 8 • 6 6 • 9 • | | | |
| E GpenEdge AppServer S OpenEdge Edit | Cr | | | |
| Sessources 🖾 📃 🗆 | 🕑 testoust.p 🔹 RunTestCust_AS.p 🕱 | P 🗄 | | |
| - → a ⊇ & マ | /* RumTestCust_AS.p: Test procedure to run TestCust. */ | | | |
| DetaTypes.html index.html OperationIndex.html | DEFINE VARIABLE SCUSAMME AS CHARACTER NO-UNDO. DEFINE VARIABLE BREVVER AS HANDLE NO-UNDO. DEFINE VARIABLE IREVUEN AS LOGICAL NO-UNDO. | | | |
| provsddoc.css runbproxygen.bat RunTestCust_AS.p Der_Setter.bat | DEFINE NEW GLOBAL SHARED VARIABLE giCallCount AS INTEGER NO-UNDO. giCallCount = 1. | | | |
| RunTestCust_Uccaup RunTestCust_WS.p RunTestCustOut_WS.p testoustout.p | CCEAIT SIEVER bServer. IReturn = hBerver:CONNECT("-AppService asbroker1 -H localbost -S 5162 -sessionModel session-fre | | | |
| WebSpeedSamples | MESSAGE "Connecting? " IReturn VIEU-AS ALERT-BOX. IF IReturn TRN Generation (ress HEIP to were stack trace) | | | |
| 🐑 과 🖃 | Status: Success on estroler11 Costoner is Test Name Calcourts is 1 | ┛ | | |
| | 🖳 Co 🛛 🖉 🖉 🖉 📴 Edge Server Monitor 🕸 🚺 🖉 📕 🔂 | - 8 | | |
| | Serve Status | | | |
| | ASBroker1Conn Started Synchronized | | | |
| | | | | |
|] 📭 🗈 | Writable Insert 1:21 | | | |

Note that the Call Count, that global variable I defined, is now equal to one. That confirms that the calling procedure and the called procedure are in fact running in different sessions. Once again, this is just a very simple example of why it's important to set up your Architect projects so that you can really test with an AppServer, because your application behavior may be different in ways much more significant than this.

Finally, let me quickly make sure you know how to provide AppServer support for a new project when you first create it. If you define a new OpenEdge project, then right on the main wizard page, there's a checkbox to **Add AppServer module**.

| OF DpenEdge AppServer - WebSamples/RunT | estCust_AS.p - OpenEdge Architect - C:\WebWorkspace | _ D × |
|---|---|--|
| m • B ≏ ∡ B % \$ • 0 | • Q ₄ • A ² • 2 • 2 • 20 • 20 • | |
| 😭 📴 OpenEdge AppServer 🐉 OpenEdge Edit | DE New OpenEdge Project | |
| Countiday depense Countiday depense Countiday depense Countiday depense Countiday Countid | By New Developer Project Image: Second and provided project Develop an Open developer Second Image: Second and second developer Second | C-UNDO. 52 -sessionHodel session-fre UTPUT oCustName). |
| 1° B | Writable Insert 1:21 | ļ |

If you check that on, then you'll be prompted to identify the server connection names to associate it with:

| C OpenEdge AppSequer - WebSampler /BunT | netfuet & n. OpanEdan Arhitect - CAWahWaykenara |
|--|---|
| File Edit Source Navigate Search Project (| DaenEdge Run Window Heb |
| 🗂 • 🗒 📥 🔏 🖦 🦦 🎄 • 📀 | • Q • A \bullet A \bullet |
| 😭 🚍 OpenEdge AppServer 💐 OpenEdge Edito | 0E New OpenEdge Project |
| Sesources 🖾 👘 🗆 | Define AppServer module |
| | Enter a name for the folder to contain AppServer source code. |
| .project | |
| | Module name: AppServerSample |
| - index.html | AppServer source folder: AppServerContent |
| OperationIndex.html | Supported servers: |
| Improved doc.css Improved doc.css | Server name |
| P RunTestCust_AS.p | ASBroker1Conn |
| P RunTestCust_Local.p P RunTestCust_W5.n | |
| RunTestCustOut_WS.p | |
| Electrostout.p | 62 -Sessionadel session-fre |
| webspectraliples | |
| Be Outline XX WE DB S D Prop | |
| Vatabler | UTPUT cCustName). |
| Variables | |
| | * • • • |
| | Publish rhannes Immeriately |
| | |
| | |
| | |
| | (?) <back next=""> Finish Cancel</back> |
| | |
| | |
|) C* E | Writable Insert 1:21 |

You can pick more than one if there's more than one server connection defined, and your project will publish to all of them. Architect will by default add an AppServerContent directory to the head of your project's ProPath. Then as for any new project you can select database connections, and other project properties. A new session starts up for your new project, if you're not using the shared AVM, and you get a message prompting you to switch to showing the project initially in the Editor perspective.

Back in the AppServer perspective, you can expand the server and see that you now have two projects associated with this server. When you create and modify code in either of those projects, they will publish the code to be runnable on this AppServer.

| 0E OpenEdge AppServer - WebSamples/Run | festCust_A5.p - OpenEdge Architect - E:\WebWorkspace | _ 🗆 × |
|--|--|-------|
| File Edit Source Navigate Search Project | OpenEdge Run Window Help | |
| 🗈 • 🔛 🖮 🔏 💷 🦦 🏇 • Ο | • Q₂ •] ∦ •] ½ + ½ + ∞ ↔ • ↔ • | |
| 😰 🚍 OpenEdge AppServer 💐 OpenEdge Edit | or | |
| Sesources 🛛 🗌 | P testcust.p P RunTestCust_AS.p 🛛 | |
| ← → @ 📄 😫 🏹 | /* RunTestCust_AS.p: | - |
| E 🖉 AppServerSample | Test procedure to run TestCust. */ | |
| BL Procedure Libraries | DEFINE VARIABLE cCustName &S CHARACTER NO-UNDO. | |
| Ann Constant | DEFINE VARIABLE hServer AS HANDLE NO-UNDO. | |
| - dbconnection | DEFINE VARIABLE 1Return AS LOGICAL NO-UNDO. | |
| | DEFINE NEW CLODAL SWADED WARTARIE as Call Course by INTECED NO DEDO | |
| - in propath | gicallouge = 1 | |
| E 🤔 WebSamples | ground in | |
| E-JL Procedure Libraries | CREATE SERVER hServer. | |
| E AnnServerContent | 1Return = | |
| bestcust.p | hServer:CONNECT("-AppService asbroker1 -H localhost -S 5162 -sessionModel session- | -fre |
| - 📄 .dbconnection 💌 | WESSAGE "Connecting? " IDeturn WIEH-AS ALEDT-BOY | |
| 🗄 Outline 🖾 🐖 D8 S 🗔 Prop 🖓 🗖 | IF 1Return THEN | |
| (A 16 D | GD0: | |
| W arishler | RUN AppServerContent\testcust.p ON hServer (INPUT 1, OUTPUT cCustName). | |
| | MESSAGE "Status: " RETURN-VALUE SKIP | |
| | | |
| | 📮 Console 🔝 Problems 🖉 Tasks 👫 Servers 🛛 🙀 OpenEdge Server Monitor 🛛 🕸 💽 🖉 📕 | 000 |
| | Server - State Status | |
| | Started Synchronized | |
| | Wheteversample Synchronized | |
| | Synchronized | |
| | | |
| | | |
| | | |
| | IL | |
|) U* 12 | | |
| | | |

That completes the two-part session on enabling AppServer support for projects in OpenEdge Architect. In other sessions I look at how you can use the AppServer as the channel for making business logic requests from a rich desktop UI such as the GUI for .NET interface you can now build right in OpenEdge.