CREATING AND DEPLOYING ABL WEB SERVICES

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 document accompanies two of the presentations that present different aspects of extending an OpenEdge application to take advantage of today's tools for building Rich Internet Applications. The first of the two sessions, entitled **Installing and Configuring the Tomcat Web Server and the OpenEdge Web Services Adapter** shows how to install the Apache Tomcat Web Server to use with the OpenEdge WebServices Adapter, in preparation for running OpenEdge procedures as services from other applications. The second session, on **Deploying and Running an ABL Procedure as a Web Service**, completes the process of exposing an ABL procedure as a Web service for access from other applications.

OpenEdge Web services are one way to invoke ABL data management and business logic procedures from a separately defined user interface. There are many choices you can make among third party Web servers that act as a Java Servlet Engine, which can then host the OpenEdge Web Services Adapter to allow other applications to run ABL procedures as Web services. One of the most popular is Tomcat, one of the Apache projects. And since OpenEdge now ships with a script to help you configure Tomcat, this document and the presentation it accompanies show you how to install Tomcat and use that OpenEdge install script to get you started.

At the URL apache.tomcat.org, you can select the latest Tomcat release from the Downloads area.



If you scroll down to see the download options, you can select the zip file from the Core list, and save that zip file to a folder on your machine where you want to install Tomcat.

Once the download is done, you need to define a couple of environment variables for the configuration script to use. In Windows Explorer, you can right-click on the **My Computer** node, select **Properties**, and then the **Advanced** tab, and **Environment Variables**.

My Computer				IX
File Edit View Favorites Tools Help			4	2
🕒 Back + 🕤 - 🏂 🔎 Search 🗞 Folders	🗟 🖉 🗙 🍤 💷 ·			
Address 😼 My Computer			💌 🄁 G	io
Folders	ystem Properties	Free Space	File System Comments	
G Desktop	System Restore Automatic Updates Remote	16.5 GB	NTFS	- 1
S My Computer	Environment Variables	19.8 GB	FAT	- 1
El Cal Disk (C:) E Disk (D:) Disk (D:)				- 1
🗉 🧝 scratch on 'Progress Development Filer (rdiservnt	User variables for john			- 1
My Network Places	Variable Value			- 1
Recycle bin My Documents	TMP C:\Documents and Settings\john\Local S			- 1
				- 1
				- 1
				- 1
	New Edit Delete			- 1
	System variables			- 1
	Variable Value			- 1
	CLASSPATH .;C:\Program Files\Java\tre1.5.0_01\lib\			- 1
	ComSpec C:\WINDOWS\system32\cmd.exe			- 1
	NUMBER_OF_P 1			- 1
	OS Windows_NT			- 1
	New Edit Delete			- 1
				- 1
	OK Cancel L			- 1
				- 1
-		-		- 1
				- 1
Cont				
4 shinche			Mu Computer	-
la colecta			J 😗 Mý Computer	_//.

The first new one you need is to establish the home directory for the Tomcat install. The code name, if you will, for Tomcat, which is also the name of a batch file you use to start and stop the web server, is *catalina*, so the script identifies the home directory as **CATALINA_HOME**. You assign this the value of the folder under your Tomcat installation directory where you will extract the zip file. The second new environment variable establishes the **JAVA_HOME** directory for the configuration file to use. And that is the jre sub-directory under wherever you installed OpenEdge.



That takes care of the two new environment variables the script needs. Once that's done you can extract the contents of the zip file. After that has completed, you need to open a batch window, which can be a **proenv** window as shown here, to run the script from. Change to the **bin** directory under the OpenEdge install.

😂 My Computer						X
File Edit View Pavorites Tool	is Help					
🌀 Back + 🕥 - 🏂 🔎 S	earch 😥 Folders 🔒	i 🕑 🗙 🍤 [
Address 😼 My Computer						💌 🄁 Go
Folders	×	Name -	Type	Free Space	File System	Comments
C Desitop	v				d TFS	
My Computer M	> Programe s a > OPENEDA	4		-	AT	
B OVD/CD-RW Dr B St soratch on Pro- the ctu Wy Network Places	ing C:\Progress\OF rrent directory is Edge102B\URK	° YENED″i∖bin to b	eginning of path and			
Recycle Bin Open Edg Proen	ge Release 10.2B a ≻od \Progress\Open	s of Mon Dec 14 Edge102B\bin	17:02:01 EST 2009			
prosnul						
					<u>.</u>	
•)	•				
4 objects						My Computer

Enter OE_TC, for OpenEdge Tomcat. This is the name of the .bat file shipped with OpenEdge that does the configuration for you. Note that this script also operates on the Tomcat install zip file directly, if it hasn't been extracted already. The script does all the work to configure Tomcat as your Web server, including establishing the OpenEdge Web Services Adapter (WSA) as a Web application in Tomcat.

The WSA is a program supplied with OpenEdge that runs as a Java Servlet in the Web server. It provides the connectivity between the OpenEdge AppServer and the procedures.



The script does all the work to configure Tomcat as your Web server, including standard Web service requests that others will use to access your ABL procedures. After confirming that you want to configure Tomcat now, enter the directory where you downloaded the Tomcat zip file.



Once the configuration script runs to completion, switch to the bin directory under the Tomcat install directory. The .bat file that starts and stops Tomcat is called catalina.bat, so to start the web server, you type catalina start. The Web server startup information comes up in a separate window, and after a few seconds, it should display INFO: Server startup in (so many) milliseconds.

Mu Consultor					
in Edit View Exweditor Tools Hele					
e cut new ratines rous rep					~
Back 🔹 🕑 🝸 🧊 🔎 Search 🎼 Folders 👘	\$ D 🗙 🎝 🛄 ·				
idress 😼 My Computer					💌 🔁 Go
olders X	Name -	Туре	Free Space	File System	Comments
Desktop By My Computer	Cocal Disk (C:)	Local Disk CD Drive National Drive	16.5 GB	NTP5	
Section (C:)	Straturur Progress Develop	Network Drive	17.0 00		
<pre>A to up_law by bocamedia j aut 239 jat sun ret to inpl_ist 239 jat sun ret to inpl_ist 239 jat sun ret to inpl_ist 230 jat sun ret to inpl_ist 230 jat sun ret ret inpl_ist 230 jat sun ret inpl_ist 230 jat sun ret inpl_ist</pre>	Here a catal lange a startup. Co Here catal lange a startup. Co latt. Native Herboddiccom latt. Native Herboddiccom Latt. De Lapse independent catal lange a startup. Herbod construction a startup. Herbod construction a startup. Herbod construction a startup. Herbodic de HITF/11 and herbodics de HITF/11 and herbodics d	And any of the first standards and a standard st	server.java wai552) thod> tingMethodR java:295) ava:433) ol start er load t	npl.	
	1 •				,

You can minimize that window, but don't close it for as long as you want the Web server to run.

Back in the **proenv** window, you can see that the **catalina** script has run to completion and confirmed some of the relevant environment variables.



The or_rc script that comes with OpenEdge is also designed to set up Tomcat for use as the Web server for WebSpeed as well, but the Apache HTTP Server is a more popular choice for WebSpeed, so this session doesn't go into the WebSpeed side of the configuration script, and the presentations in this series that focus on WebSpeed use the Apache Web server.

In any case, you can quickly verify that Tomcat is running. Unless you configure it otherwise, it runs on port 8080, so if you enter that port number on localhost as a URL, you see a confirmation page downloaded from the Apache site that the web server is set up successfully:



If you then add wsa and the default OpenEdge Web Services Adapter wsa1 to that URL, you get a confirmation message from the WSA that you're communicating with it as well:



Now that Tomcat is installed and configured, you can go back into OpenEdge Explorer and take a look at the properties for the default Web Services Adapter, wsa1. If you expand the Web Services Adapter group for OpenEdge and select wsa1, then under **Configuration**, you can see the URL used to access the WSA.

C / //or albost-9090 / fathom htm - 1	Windows Internet Explorer		
Carlos - Chttp://localhost:909	Difathom.htm	V R 44 X R Goode	- 0
File Edit View Favorites Tools × Google	Heb 💽 🚰 Search • • 🛷 • 🌗 • 🔯 Shar	re * 🚳 * 🗆 * 🔍 Sidewiki * 🎽 Oteck * 👪 Translate * 🏷 Auto	sFill • 🗞 • 🌛 🔍 • 🔾 Sign In •
🚖 Favorites 🛛 🙀 🏀 Suggested Site	s 🔹 🙋 Free AOL & Unlimited Internet 👔 Fr	ee Hotmal 🔊 Web Sice Gallery -	
()/localhost:9090/fathom.htm		💁 • 🖸 - 🗁 👼	• Page • Safety • Tools • 🚷 • '
admin on NBJOHNXP1 (contain XResources ✓Options ?H	ners: 1, offline: 0, unknown: 0) elp		© Ø Ø
Sort by: Container	Web Services Adapter Con	figuration: nbjohnxp1.wsa1	9. Disabled (6 Minutes) Not Polled Poll Count: 0 Fail Count: 0 (0.0%)
 ■ ■ nbjohnxp1 ■ OpenEdge ● ● AppServer Internet 	Edit	DL Logging Setting Security Advanced Features	
🗉 📵 App Server	Location		
 Batabase Messengers Oname Server 	Location: URL: Web Service Protocol Type:	local http://NBJOHNXP1:8080/wsa/wsa1 Duplicate WSA Url value	
 SonicMQ Adapter Web Speed Web Services Adapter 	Admin Role Links		
4 , wsa1	AdminRole.PSCAdmin AdminRole.PSCOper		
· · ·	1012		
Done		Local intr	anet 🖗 • 🔩 100% •
		,,,,,,,,	,

You can verify that it's set to 8080, or whatever port number your web server uses, and also set more advanced properties. Back on the main wsa1 page, you can then select the **Control** link, and confirm that the WSA is enabled and active.

//localhost:9090/Tathom.htm - V	Vindows Internet Ex	plorer			<u>_0 ×</u>
G v D http://localhost:9090	1/Fathom.htm		- 8 4	🗙 🚼 Google	ρ.
File Edit View Favorites Tools	Help				
× Google	💌 🔧 Search = 🕬	🧭 • 🗣 • 🔁 Share • 👰 • 🖘 🖉	Sidewiki * 🍧 Check * 👪	Translate • 🎦 AutoFill • 🖄 • 🌛	🔍 🔹 🔵 Sign In 🔹
🔆 Favorites 🛛 🍰 🏀 Suppested Site	s 🔹 🙋 Free AOL & U	nimited Internet 🙋 Free Hotmail 👩 Web	Sice Gallery •		
()/localhost:9090/Fathom.htm			6	🔹 🔂 🗉 🖶 🔹 Page - Safet	y • Tools • 🔞 • 🏾 »
admin on NBJOHNXP1 (contain XResources ✓Options ?H	ners: 1, offline: 0, elp	unknown: 0)			000
Sort by: Container	Web Serv	ices Adapter Control: nbjohnxp	1.wsa1		
B nbjohnxp1 OpenEdge	Edit				
App Server Internet	Adapter summar	у			
App Server	Adapter name:	wsa1			
Database	Host:	NBJOHNXP1			
* Messengers	Status:	ACTIVE http://NR.IOHNXP1:8080/wsa/wsa1			
SonicMQ Adapter	Properties	http://tooornou.coopingenour			
Web Speed	Enabled				
Web Services Adap	Lindbird				
A WSaT		Progress S	oftware Corporation (www.progress.	(1000	
1 ×					
9*+	< 9 % ?				C. (4.191)
Done				Uocal Intrianet	A * 100% * //



In the second part of this two-part session I show how to deploy an ABL procedure as a Web Service and invoke it from another ABL test procedure. I start with about the simplest possible ABL procedure, called testcust.p. This is also used in a similar example from another session to verify that the OpenEdge default AppServer is running properly.

```
/* 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 takes a customer number as input, and passes back a string as if it were a customer name from the database, along with a **RETURN-VALUE** from the ABL **RETURN** statement.

So how do you set this up to be called as a Web service? First, you need to make your ABL procedures accessible through an OpenEdge AppServer to make them callable as Web services.



The AppServer is designed to handle requests that come directly from OpenEdge clients. To adapt it to receive requests as Web services, you need to insert a Web server into the mix. As shown earlier, this example uses the Apache Tomcat Web server. The Web server holds the Web Services Adapter (WSA), a program supplied with OpenEdge that runs as a Java Servlet in the Web server. The OpenEdge Proxy Generator, ProxyGen for short, generates the Web Service Application Descriptor (WSAD) and a test WSDL (Web Service Descriptor Language document) to use in deploying and testing the service. At runtime, a request comes in from a Web service client into the Web server, in the form of a SOAP request. The request is routed to the AppServer and your ABL procedure, and output returned to the caller as another SOAP message. All the message construction and formatting is taken care of for you.

You start ProxyGen from the OpenEdge Start group, and under the File menu, select New.



What you're defining is called an *application*, and for a non-persistent procedure -one without multiple named internal entry points -- you define an **AppObject**. The example uses one called testcust. You need to adjust the Propath so that the compiled ABL code will be found, which is here in the WebSamples folder. Next you need to define what Procedures are part of the AppObject.

OF OpenEdge Editor - WebSamples/tes	cust.p - OpenEdge Architect - C:\WebWorkspace	_ [] ×
File Edit Source Navigate Search Pri	vject OpenEdge Run Window Help	
] 📑 • 🗟 🖄 🖌 📖 🗞 🎋	• Q • Q •]	
😭 🎳 OpenEdge Editor		
Sesources 23	D Internation X	- 0
6 - + 6 E 5	Proxy Generator for Open Clients - untitled	A
Tunterovygen bat	File SubAppObject Procedure Tools Options Help	
RunTestCust_AS.p	🗈 📾 🚨 🙏	
RunTestCust_Local.p		
RunTestCust_W5.p	AppUbject Proradures	
RunTestCustOut_WS.p	Name Select the Procedures tab	
e testoustout o	Protath	
E 🥵 WebSpeed	Components New	
JL Procedure Libraries		
	Edt.	
.project	Delete	
E HORCACOURTING	Description	
Securitiene 23 📅 DB 5 🗔 Prop		
10. 1		
(i) (ii) Variablar	Y I	
	Enable Unknowns for Parameter and Return Values	
	Map Temp-Table Parameters to SQL ResultSets for Java	
	Write ProDataSet Before-Image in XML	
		1
		×
	🖾 Console 🕴 🗄 Problems 🖉 Tasks 💿 🗮 🙀 👘 🛃 👘 🛃 👘	
T* 12	Writable Insert 1:1	

There's only one, and it's a non-persistent procedure, so you select **Add**, and the r-code for the test procedure:

OF OpenEdge Editor - WebSamples/testcusta File Edit Source Navigate Search Project	p - OpenEdge Architect - C/WebWorkspace OperEdge Run Window Heb		-0×
] 🗂 • 🗉 🖾] 🔏 💷 🦦] 🌼 • O	• 💁 •] 🖋 •] 🖗 • 🖗 • 🏷 🗢 • 🗠 •		
🔛 🎳 OpenEdge Editor			
🥰 Resources 😂 📃 🗆	Prestruction 🗙		
	🛜 Add Non-persistent Procedures for testcust		1
Trunbproxygen.bat	Propath Selection D.WebWorkspace\WebSamples\	Use Path in Name	
P RunTestCust_AS.p	File Filter	Perform Recursive Add	
- RunTestCust_WS.p	Folders		
RunTestCustOut_WS.p	C C:\WebWorkspace\WebSamples\	Add p.	
testcust.p testcustout.p	BunTestCust_Localr	Close	
🖻 👹 WebSpeed	BunTestCustOut_WS.s		
decorpection	E] testquat.r	Help	
.project	Select the tes	tcust.r file	
- in propath			
Showcosconers.nom			
😹 Outline 🕮 📅 DB S 🗔 Prop			
(O) 🗄			
🗈 🔇 Variables			
	1 directories selected	0 Procedures	
		m h	
			-
			×
	📮 Console 🕴 🚺 Problems 🕗 Tasks	- X 💥 🕞 🛃 💭 🧭 ピ - 😂 -	· • •
J 🕐 🗈	Writable Inser	et 1:1	

You add that to the AppObject. One more thing you need to do in this example is allow for the ABL RETURN-VALUE that testcust.p uses. This is under the **Customize** option for the procedure.

OF OpenEdge Editor - WebSamples/tesi File Edit Source Navigate Search Pro	Rcust_p-OpenEdge Architect-CL\WebWorkspace oject OpenEdge Run Window Heb	_0×
] 🔁 • 🛛 🗠] 🔏 📖 🦦 🏇	• O • Q •] -] - [-] - [-] - [-]	
🗈 🗳 OpenEdge Editor		
Sesources 23		° 0
$\leftarrow \rightarrow \sim \equiv 1$	Proxy Generator for Open Clients - unotited	<u> </u>
runbproxygen.bat RunTestCust_AS.p		
RunTestCust_Ucca.p RunTestCust_W5.p RunTestCustOut_W5.p	AppDigect Procedures	
testoust.p testoustout.p WebSpeed JL Procedure Libraries	Non-persident Procedures (1) Keynetak cartooraal values (1) Keynetak cartooraal values (1) Cartooraal values (1	
dbconnection project	Ramove Select the Customize menu item	
ShowCustomers.html	Move to Persistent Move to Non-persistent	
E Outine 23 PDB 5 Prop	Perintert Procedures (0)	
	z	<u>ح</u>
	😑 Console 🕴 🚹 Problems 🖉 Tasks 💿 🖉 🖓 😰 😁	• 📑 • 🗂 🗖
J 📬 🖪	WrRable Insert 1:1	

Check on the option to return the ABL **RETURN-VALUE**. It can't be returned implicitly, as it is from one ABL procedure to another procedure that calls it directly, so it will be passed as an additional output parameter called **result**.

OF OpenEdge Editor - WebSamples/test File Edit Source Navigate Search Pro	cust.p = OpenEdge Architect = Ci_WebWorkspace Ject OpenEdge Run Window Help	<u>_0×</u>
] 📬 • 🗟 🖻 🗹 💷 🗞 I 🏇 •	• Q • Q •] Ø •] 9 • 7 • 9 • 9 • • •	
📑 🤹 OpenEdge Editor		
Personal 32 Index-regent bit Personal 20 Persona 20 Persona 20 Person	Constrained Const	0 L
] 📭 🗈	Writable Insert 1:1	

Now you need to generate the proxy for this new AppObject.

OF OpenEdge Editor - WebSamples/test	Rcust.p - OpenEdge Architect - C: WebWorkspace roject: OpenEdge Run: Window: Heb	=IOI×I
1 1 · 1 · 1 / 1 · 1 · 1 · 1 · 1 · 1 · 1	• 0 • 9 • 1 4 • 1 9 • 5 • 9 • • • •	
C OpenEdge Editor		
Resources 13		
	🛜 Proxy Generator for Open Clients - untitled	-
Tunboroxygen.bat	File SubAppObject Procedure Tools Options Help	
- RunTestCust_AS.p	New Cort+N	
- RunTestCust_Local.p	CONFO CONFO	
RunTestCust_WS.p	Generate	
Pontescoscod_ws.p Petrust.p	Save Solort the Concrete monuitors	
e testcustout.p	Save As Select the Generate mend term coutr	
🖻 👹 WebSpeed	Close	
It Procedure Libraries	C:\WebWorkspace\WebSamples\testcust.xpxg	
- x project	C:\WebWorkspace\WebSamples\testoustout.xpxg	
propath	Ext	
- ShowCustomers.html	Move to Persistent Move to Non-persistent.	
	Persistent Procedures (0)	
ge outine 23 me be s i Prop		
(0.3		
Variables		
		-
		>
	😑 Console 🕮 👔 Problems 🧟 Tasks 💿 🗧 💥 🎇 🕞 🖉 ピ ピ - 🗂 -	
) T* 🗈	Writable Insert 1:1	

This creates all the pieces reviewed in the diagram of what a Web service call requires. ProxyGen can create proxies for many different kinds of external access to an OpenEdge session, but this one is a Web service.

Construction of the same of the sector of the sector here the same here the sector here the same of the sector here the same of the sector here the same of the sector here the same sector here	Lgs Openstäge Architect - Cl VerblWorkspace OpenSigs Ren. Worken Heip Que J Art J S + Q + S → Art Art Art Architect Disconstruct: Wards Concelled for Open Clearly - wateliked Statig-Clearly Total on Openson Heip Concelled Fronted for Test Cattors Clear Day Series (Jacks - Series Web Series) Series Native Clear Day Series (Jacks - Series Veb Series) Series Native Clear Day Series (Jacks - Series)	
Bestack a Bestack a	Oter Play / Serve Definition to Elemente NET Vivé Serve Vivé Serve Serve Native Innocation Java Sore: Web Serve Adobinite Innocation Vieto Innocation Vieto Innocation Paren Alle Play / Serve Vieto Innocation Paren Alle Play / Serve Vieto Innocation Paren Alle Play / Serve	Corell Help
] C° 🖻	Writable Insert 1:	1

The procedure is deployed where the standard AppServer, **asbroker1**, can run it, so you need to specify that here. You tell ProxyGen where the generated output should go, and enter any values for the **Author** and **Version**. You want the service to run without binding the AppServer session it gets run on, so make sure the **Session Model** is set to **Session-Free**.



Now you enter information specific to the Web Service proxy type. Every service has to have a **namespace** that uniquely identifies it. In this example I use one that identifies it as a Progress service, under my username. This becomes part of the identification of the service when it's called. You also need to specify the **URL** that identifies the port number the Web server will be running on.

DE OpenEdge Editor - WebSamples/test File Edit Source Navigate Search Pro I S III Survey I S III Survey IIII I S IIII Survey IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	ankp-Opendage Architect -C.\WebWarkspace (ct. CoperEdge Fun. Window Heb • $\mathbf{O} + \mathbf{Q}_{\bullet} + \int \mathbf{A}_{\bullet}^{\bullet} + \int \mathbf{A}_{\bullet}^{\bullet} + \int \mathbf{A}_{\bullet}^{\bullet} + \mathbf{A}_{\bullet}^{\bullet} + \mathbf{A}_{\bullet}^{\bullet} + \mathbf{A}_{\bullet}^{\bullet}$	_ [] X
Approximation Approxi	Concertain California Concertainty Concertainty	
] T° 🗈	Writable Insert 1:1	

Since this example uses Tomcat as the Web server, that runs on port 8080. And the service will be deployed on the default Web Services Adapter for OpenEdge, wsa1.

There's no specific **SOAP Action** needed, so that can be left set to **Blank**.

The **WSDL Style** tells ProxyGen which of several XML formats to use for the parameters to the request. The **Doc/Literal** style uses a standard XML format to represent all of the parameters as a single XML element. If you choose this style, ProxyGen does the necessary formatting for you, and the Web service consumer is informed that this is the style to expect, so there's no work for you to do. This is generally the recommended choice.

By default ProxyGen furnishes a suffix of Obj foreach **PortType** name -- representing a set of related operations that the Web service supports -- and **Service** for each service name, so you can leave those as they are.

The **Generate Test WSDL** option gives you a WSDL definition document at development time that you can inspect (using the WSDL Analyzer). You tell ProxyGen where you want it to save the files it creates, and it's done.

The State Source Hendles Hen	
Constant of the second se	
Coverdage Edar C	-
Constrained and a second of an analysis of the formation of the forma	-
Areauxies to Areauxi	×
Concerned and a second a sec	4
Provident barles Prov	
BunferdLock_AS BunferdLock_AS BunferdLock_VS BunferdL	
Augradian Sector (Constant) Augradiant Management (Median Sector (Constant)) Augradiant (Median Sector (Median Se	
Publication 24, 395 p Brandford Look 24, 395 p Bran	
Bestranden B	
C. Weby only part Web sampler Methods 1 Compared to the sampler Methods and the sampler Metho	
Construction C	
B J. Procedure Libraries Procedure Libraries Advanced	
disconnection disconnection provide p	
Procyclastic and a successed in	
ShowCustomers.html Proxies can be found in C:\WebWorkspace\WebSamples\	
Coutine 🛛 🖻 DB S 💷 Prop	
(b) 3	
🗄 🚫 Variables	
Provide Generation Surgeorded	
	8 F 🗍
🕒 Console 🛛 🦹 Problems 🖉 Tasks 💿 🗰 💥 🍇 🔒 🖉 🦉 ピ 🖬 - 😭	• •
P E Webbe Insert 1-1	

So that's the job ProxyGen does, creating the descriptor files the Web server will use to route a request to the right AppServer and the right ABL procedure. OpenEdge Explorer reads the Web Service Mapping file that is part of what ProxyGen created, and uses this to deploy the procedure as a service.



To examine the AppServer and Web Services Adapter, open OpenEdge Explorer from the OpenEdge start menu, expand the OpenEdge group, and the list of AppServers. The example specified in ProxyGen that it would be accessing testcust.p as a service from the default broker, asbroker1, so you can just confirm that it's active. If it isn't, you could activate it here.

Under the Web Services Adapter group, you can select the default Adapter, wsa1, and activate it if needed. Next you need to deploy the new Web service to wsa1. To do this you select the **Deploy** link. The one value you have to fill in is the Web Service Mapping (WSM) file name that came from ProxyGen. There's no Browse button to use to locate the file, so you have to type in the pathname, and submit the change.

C//localhost:9090/fathom.htm - Wind	ows Internet Explorer			
Go v Dhitp://locahost:9090/Fath	iom.htm		💌 🗟 🖘 🗙 🔀 Google	۶.
File Edit View Favorites Tools Hel × Google	9 🛃 Search • • 🔊 • 🐢 🛛 🖸	Share • 🚳 • 🗆 • 🔍 Sidewiki •	🍜 Check - 👪 Translate - 🎦 AutoFill - 🖓	💊 - 🌛 🔩 - 🔾 Sign In -
🙀 Favorites 🛛 🙀 🏉 Suggested Sites 👻	Pros ACL & Unimited Internet	Pree Hotmail 😰 Web Sice Galery	•	
@//localhost:9090/fathom.htm			🚹 • 🔂 - 🖂 👼 • Page	r - Safety - Tools - 😥 - 🏾 »
admin on NBJOHNXP1 (containers: Resources COptions PHelp	1, offline: 0, unknown: 0)			88
Sort by: Container	Web Services Ad	apter: nbjohnxp1.wsa1		
Binbjohnxp1 GopenEdge AppServer Internet Ada GopenVer	Subret Cancel Deple Click the Submit Define a new Web Service for ProxyGen.	t button the selected Web Services Adap	pter. You must provide the Web Service Maj	p file generated by
esbbroker1	WSM File (path and Nar	ne): C:\Web'	Workspace\WebSamples\testcust.wsm	
icfepos ion construction ion construct	s Progress Software Corporation (<u>trace anappens aucri</u>) apter ss Adapter			
× • •	₹ 3 % ?	a ucal		in • 10 1005 •

You can then review the deployment information, which is everything that was specified in ProxyGen.

🕽 🕑 🗢 🕲 http://localhost:9090/Fat	hom.htm	<u>∎</u> 🗄 🐄 ×	🕻 🎦 Google 🧖
le Edit View Pavorites Tools He	lo .		
Google	🚽 🚼 Search 🔹 🧭 🕈 👘 🔹 🔯 🕄	Share * 🔕 * 🗆 * 🤍 Sidewiki * 👘 Check * 🜆 🕅	lranslate + 🎦 AutoFill + 🔏 + 🌛 🛛 🔩 + 🔵 Sign 1
Favorites 👍 🍘 Suggested Sites 🔹	nee AOL & Unlimited Internet	Free Hotmail 🙋 Web Sice Gallery 💌	
) //localhost:9090/fathom.htm		9	• 🔯 - 🖂 🛞 • Page • Safety • Tools • 😧 •
admin on NBJOHNXP1 (containers	a 1, offline: 0, unknown: 0)		S S
Resources 🗹 Options 🖓 Help			
Resources	Web Services Ada	enter phioboxo1 wsa1	
art hy Container	(barry) Web Services Ada	pter. nojomiti pr.wsa i	
on by 1	Coloria Connel 1		
🗄 🔂 nbjohnxp1	Submit Cancer		
OpenEdge	Click the Submit	button	
App Server Internet Ada	Web Edenomo:	C:WebWekepaceWebSamplecitectoust war	
App Server	Name:	testcust	1
esbbroker1	Name:	testcust	
icfrepos	Web Service Namespace:	urn:services-progress-com	
🗆 📄 Database	SOAP Action:		
📧 🍬 Messengers	Append to SOAP Action:		
📧 🥥 Name Server		C RPC / Encoded	
SonicMQ Adapter	WSDL Style/Use:	O RPC / Literal	
🗈 🕕 Web Speed		Occument / Literal	
Web Services Adapter		Proven Selbran Contraction (new proven	
Woor		Progress Software Composition (Contractory Contractory	(dom)

Pressing **Submit** again, you see that Explorer confirms that the service has been deployed, so it's ready to enable.



Scrolling down in the wsa1 page, you can see a list of all the deployed services. If there's more than one deployed service, you can select one from the dropdown list and click **Select**. Then you click the **Status Enablement** link.

Carlos and hits // los about 0000 if all	om bitm	To the V Results	P
S S A B web the carest solution	on non		
File Edit New Pavarites Tools Hel			
× Google -	🛂 Search + 🔶 + 🤹 + 🔁 Share + 🥸 + 🗔 + 🔤 Sk	lewiki * 🛛 💝 Check + 👪 Translate + 🎦 AutoFill * 🐴 * 🍐	🗸 • 🕞 Sign In
Favorites 🔗 🖉 Suggested Sites •	Pree AOL & Unlimited Internet 😰 Free Hotmail 😰 Web Sice	Galery •	
B //incolored 0200 Saltors htm		A + C + C - A - Pare + Set	ety + Tools + 🕥 +
All horan our south as fear than			
admin on NBJOHNXP1 (containers:	1, offline: 0, unknown: 0)		9 9 9
Resources Options 2 Help			
Resources	Case Login	The off	unginer
Sort by: Container *	Ligit Ser WSA Web Server usemame/password	Log off of the WSA Web Server	
- Bablahavat		Deploy	
E III OpenEdge	List the deployed web services	Deploy a new web service	
AppServer Internet Ada	Import import web service	Break Set web service defaults	
= @ App Server		alast	
asbroker1	Deployed Web Services for testcust lesicust =	and I	
esbbroker1	Status Enablement	Defana Cipciate a deployed web service	
Database	Applicat	(som Export	
• Messengers	Prass Deployed Select the Status Enablement	INK Export a deployed web service	
🖲 🥥 Name Server	Statistic	Undeploy	
* SonicMQ Adapter	Deployed web service run-time statistics	Underly Underloy a web service	_
• 🛈 Web Speed	Operations views		_
Web Services Adapter	Web Service Adapter status information	State Web Services Adapter run-time alatistic informati	on
 wsa1 	Run-time Properties		
	Press Web Services Adapter run time properties		
L	Progress Soft	wate Corporation (www.progress.com)	

You then click **Enable** to enable it to be run as a Web service.



This sequence shows the role Explorer plays: to read the output from ProxyGen and use that to set up the procedure to be run on an AppServer, accessed through a Web server using the OpenEdge Web Services Adapter.

Now I'll show a simple example of how to get help in writing a procedure that will invoke the Web service. Of course, you wouldn't normally run an ABL procedure from another ABL procedure as a Web service – you could just run it directly on the AppServer – but this example does that just to test that it works. In ProxyGen, I had checked the option that said to generate a test WSDL, the descriptor document. You can use the test WSDL to help code a procedure that runs the Web service.

To access the test WSDL document, you can start a **proenv** window from OpenEdge, and set your directory to where all the generated proxy elements are. There's an OpenEdge utility to interpret a WSDL document for you and generate sample code and documentation, and it's called: **b**-for-batch **pro**-for -progress **wsdl** and **doc** for documentation, **bprowsdldoc**, and it takes the WSDL document that came out of ProxyGen as an argument. This is the same tool you would use to analyze the WSDL that describes a non-OpenEdge service that you wanted to run from an ABL application. The utility generates a set of HTML files that provide detailed documentation of the service for you. The top-level file is always named index.html, so if you generate more than one of these for different services, make sure to put them into different directories.

DenEdge Editor - WebSamples/testcust.p	o - OpenEdge Architect - Ci-WebWorkspace	_O×
E? #* OpenEdge Editor		
Kesources 23	pDLC: C:\Progress\OPENED~1	
← → ⊗ ⊟ & ▼	Inserting C:\Progress\OPENED~1\bin to beginning of path and the current directory is C:\OpenEdge1025.WBK	~
runbproxygen.bat RunTestCust_AS.p	OpenEdge Release 10.2B as of Mon Dec 14 17:02:01 EST 2009 preeu⊁cd \WebWorkspace\WebSamples	
RunTestCust_Local.p RunTestCust_WS.p	proenv>hprowsdldoc testcust.wsdl Documentation written to	
estcust.p	file:///C:\WebWorkspace\WebSamples/index.html	
e testcustout.p	proenv>	
B-JL Procedure Libraries		
- sproject		
	pousoname - "No name".	
E Outline 23 MT DB 5 Prop	RETURN "Failure!".	
(i) 30 E	END.	
🗄 🔍 Variables		
		_¥
) 0 ° E	Writable Insert 3:1	

If you open index.html, it comes up in your browser. Remember that the call that's defined in my application is called a **Port Type**, and it has obj as a generated extension.

🖉 WSDL - Windows Internet Explorer		
🕞 🕘 🔹 🕼 C:\WebWorkspace\WebSamples\service_um_services_progress_com_john_testcust_testcustService.html 💿	🔹 🔀 Google	P -
File Edit Wew Favorites Tools Help		
🗴 Google 💽 🛃 Search + 🖗 🚳 + 🚔 + 🔯 Share + 👰 + 🗔 + 📮 Sidewiki + 🦃 Check +	🧦 Translate * 🎦 AutoFill * 🐁	🕗 🤌 • 🔵 Sign In •
🔆 Favorites 🛛 🍰 🍘 Supported Sites 🔹 🐔 Free AOL & Unimited Internet 💰 Free Hotmail 💰 Web Site Gallery 🔹		
😕 - 🕲 //localhost:9090/Fathom.htm 🏾 🏉 WSDL 🗙	🏠 • 🔂 - 🖃 🌧 • Page •	Safety • Tools • 🕡 • 🏾
		Operation index
WSDL		
Author=John, EncodingType=DOC_LITERAL, Proxygen_Product=Progress Version 10.2B		
Location		
file////C:\WebWorkspace\WebSamples\testcust.wsdl		
Target namespace		
um:services-progress-com;john:testcust		
testcustService service		
No documentation found in WSDL.		
Port types (persistent procedures)		
testcustQbi No documentation found in WSDL.		
Data type Select the testcustObj link		
File:///C:/WebWarks_progress_com_john_testcust_testcustObj.html	My Computer	👘 • 🗮 100% • //

If you select the testcustObj link, you see detailed documentation for the service. Scrolling down, there's an example of how the procedure testcust would look as an internal procedure in my **AppObject**, with its parameters. The result output parameter represents the ABL RETURN-VALUE that I specified. And remember that in reality, testcust is an external .p, so calling it won't bind the AppServer session it runs on; it's just represented as an entry point in the Web service.

Port type: testcustObj - Windows Internet Explorer	
🌀 🕞 💌 🖉 C:\WebWorkspace\WebSamples\portType_urn_services_progress_com_john_testcust_testcustObj.html 🔄 🐓 🗙	Google 🖉 •
File Edit View Feronites Tools Help 🗴 Google 🔄 🛃 Search • 🕂 🌮 🏰 🔯 Share · 👰 • 🗆 • 🗍 Sidewidi • 👹 Orieck • 🔐 T	ranslate = 🍗 AutoFill = 🐁 - 🌛 🛛 🔩 = 🔵 Sign In =
🖕 Favorites 🛛 🙀 🏉 Supposted Sites 🔹 🔊 Free ACL & Unimited Internet 🖉 Free Hatmail 👩 Web Site Gallery 🔹	
😳 🔹 🕲 //localhost:9090/fathom.htm 🛛 🏉 Port type: testcustObj 🗙 🚺	• 🕤 - 🖃 🖶 • Page • Safety • Tools • 😥 • 🎙
Operations (internal procedures)	
NOTE: Some of the procedures documented below were documented differently in the 10.0 releases of Op clearts written for release 10.0, or are deploying web service clearts in a mixed 10.0 post-10.0 installation, re option.	enEdge. If you are maintaining web service -execute this utility with the -showl00style
PROCEDURE <u>Learcours</u> : DEFINE INVIT FRAAMMETER pCustNum AS INTEGER NO-UNDO. DEFINE OUTFUT FRAAMMETER result AS CHARACTER NO-UNDO. DEFINE OUTFUT FRAAMMETER pCustName AS CHARACTER NO-UNDO. EDE FROCEDURE.	No documentation found in WSDL.
Note: this operation may also be called as a FUNCTION. See the operation detail for the function signature	
Connection details Connection parameters	
-MSDL 'testcust.vsdl' [-MSDLDderId userid] [-MSDLBasyood purwowd]	
[Service testcustService [ServiceNamespace un:services-progress-com:john:testcust]	
lone	S My Computer

Scrolling down further, there are sample ABL statements for creating a server instance to connect to the Web service, connecting to the service, and running the testcust entry on that server. All this is very parallel to making an ordinary AppServer call.

Port type: testcustOb) - Windows Internet Explorer
🌀 🕤 💌 😰 C: (WebWorkspace) WebSamples/portType_um_services_progress_com_john_bestrust_bestrust_bestrustObj.html 💽 🍕 🗙 🕌 Google 🖉 -
File Edit View Favorites Tools Help
🗴 Google 🔤 🛃 Search + 6 🧭 + 🕋 + 🔯 Share + 👰 + 🖘 - 🖓 Sidewili + 🚏 Check + 👪 Translate + 🐚 AutoFili + 🐁 + 🌛 🔌 + 🕟 Sign In +
🁷 Favorites 🙀 🍘 Supposted Sites 🔹 🙋 Free AOL & Unlimited Internet 🙋 Free Hotmail 🙋 Web Site Gallery 🔹
🕆 🛱 //locahost:9090/fathom.htm 🏾 🏉 Port type: testoustOb) 🗙 👘 + Page + Safety + Tools + 👔 +
Example
This example shows the steps needed to prepare for calling one of the operations described below.
DEFINE VARIABLE hNebService AS HANDLE NO-UNDO. DEFINE VARIABLE htestcustobj AS HANDLE NO-UNDO.
CREATE SERVER hNebService.
hNebService:CONNECT("-WSDL 'testcust.wsdl'").
RUN testcustObj SEI htestcustObj ON hWebService.
Operation (internal procedure) detail
Top Service Data types Operation index
testcust
N. Lawrence A
No accumentation Jouna in WSDL.
Procedure prototype
PROCEDURE testoust:
one 🛛 🗍 🛃 My Computer 🚓 + 💏 100% + -

Looking down further, the example shows both the code you would write to invoke the service as if it were a function and as if it were an internal procedure, even though it's actually a stand-alone .p.

🖉 Port type: testcustObj - Windows Internet Explorer	_ 0 ×
🚱 🔄 💌 🔊 C: (WebWarkspace) WebSamples port Type_urn_services_progress_com_john_testcust_testcustObj.html 💿 😔 🗙 🚼 Google	P -
He Edt Wew Favortes Tools Heb X Google 💽 🛃 Search - 🖓 🗇 😭 Share - 🔯 Share - 💭 Schenki - 🍏 Check - 🔛 Translate - 📡 Azofii - 🐁 - 🎸	🧳 • 🖂 Sign In •
👷 Forvarites 🛛 🎪 🍘 Suggested Sites 🔹 🙋 Free ACL & Unlimited Internet 🥐 Free Hotmail 🙋 Web Site Galery 🔹	
🔠 🔹 🔯 //localhost:9090/fathom.htm 🛛 🏉 Port type: testcustObj 🗙 👘 🗧 🖄 - 🖄 - 🖄 👘 - Page - Saf	ety • Tools • 🔞 • 🤷
Example	-
DEFINE VARIABLE prounthm AS INTERES NO-NDRO. DEFINE VARIABLE prount AS CHARACTER NO-NDRO. DEFINE VARIABLE prousthame AS CHARACTER NO-TORO.	
FUNCTION testoust RETURNS CHARACTER (INFUT poustNum AS INTEGR, OUTFUT poustNum AS CHARACTER) IN htestoustOb).	
<pre>/* Function invocation of testcust operation. */ result = testcust(pCustNam, pCustName).</pre>	
/* Procedure invocation of testoust operation. */ RUN testoust IN htestoustObj(INFUT pCustNum, OUTFUT result, OUTFUT pCustName).	
Parameters	
pCustNum	
The parameter can be the unknown value (?).	
This value is defined as an XML Schema int value.	
Done V Computer	

So based on sample code like what the WSDL documenter tool generates, you can create a test procedure such as the following to run testcust from an OpenEdge client:

/* RunTestCust_WS.p: Test procedure to run TestCust.p as a Web service */ DEFINE VARIABLE pCustNum AS INTEGER NO-UNDO. DEFINE VARIABLE result AS CHARACTER NO-UNDO. DEFINE VARIABLE pCustName AS CHARACTER NO-UNDO. DEFINE VARIABLE hWebService AS HANDLE NO-UNDO. DEFINE VARIABLE hTestCustObj AS HANDLE NO-UNDO. DEFINE VARIABLE lStatus AS LOGICAL NO-UNDO. DEFINE VARIABLE lStatus AS LOGICAL NO-UNDO. DEFINE NEW GLOBAL SHARED VARIABLE giCallCount AS INTEGER NO-UNDO. giCallCount = 1.

Here's the **CREATE SERVER** statement, and the **CONNECT** statement with the full identifier for the service in its namespace.

```
CREATE SERVER hWebService.
lStatus = hWebService:CONNECT("-WSDL
'http://localhost:8080/wsa/wsal/wsdl?targetURI=urn:services-progress-
com:john:testcust'").
MESSAGE "Connected? " 1Status VIEW-AS ALERT-BOX.
IF 1Status THEN
DO:
    RUN testcustObj SET hTestCustObj ON hWebService.
    MESSAGE "After RUN TestCustObj, handle is " hTestCustObj VIEW-AS ALERT-BOX.
    RUN testcust IN hTestCustObj(INPUT pCustNum, OUTPUT result, OUTPUT pCustName).
    MESSAGE "Status: " RESULT SKIP
            "Cust Name is " pCustName SKIP
            "Call count is " giCallCount
           VIEW-AS ALERT-BOX.
   hWebService:DISCONNECT().
END.
DELETE OBJECT hTestCustObj NO-ERROR.
DELETE OBJECT hWebService NO-ERROR.
```

When you run the test procedure, you get connected to the Web service, then see a message confirming that you got a valid handle back, representing the running instance of the service, and then get back the expected output values:



The call count of 1 verifies that my calling procedure and the called procedure are running in separate OpenEdge sessions, so they're looking at different copies of the global variable that's incremented in both places.

In conclusion, although there are a number of steps involved in setting up a procedure to be invoked as a Web service, ProxyGen does most of the work for you, and OpenEdge Explorer lets you easily deploy the services you create on an OpenEdge Web Services Adapter. Once they're set up, any other application running in any language on any platform can invoke them over the Internet. That's the value of our Web services support in OpenEdge.