



IBM® WebSphere® SDK for Web Services (WSDK) Version 5.1

“The integrated developer kit for Web services”



WSDK 5.1 at a Glance

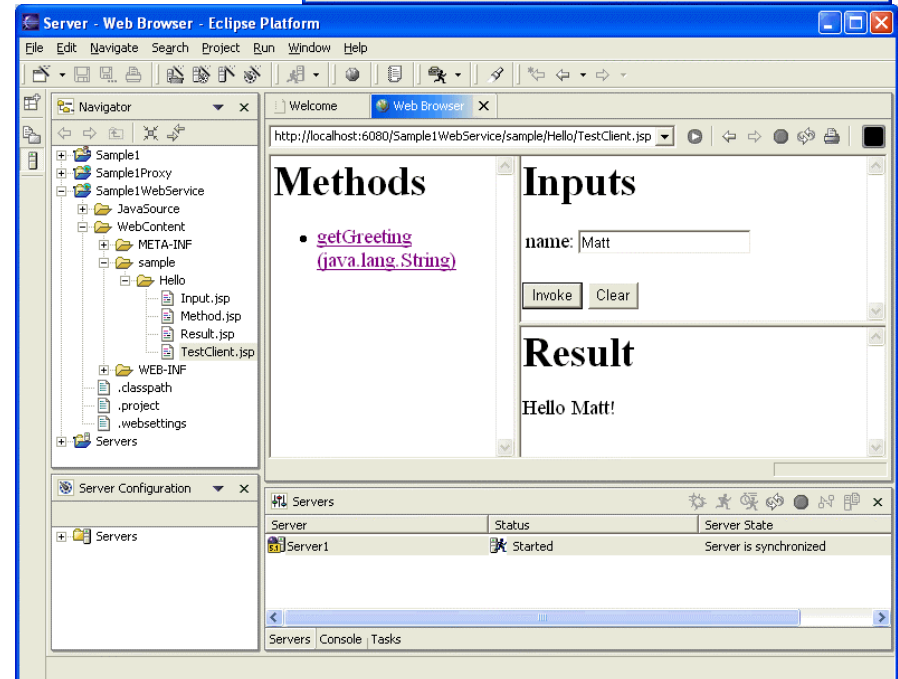
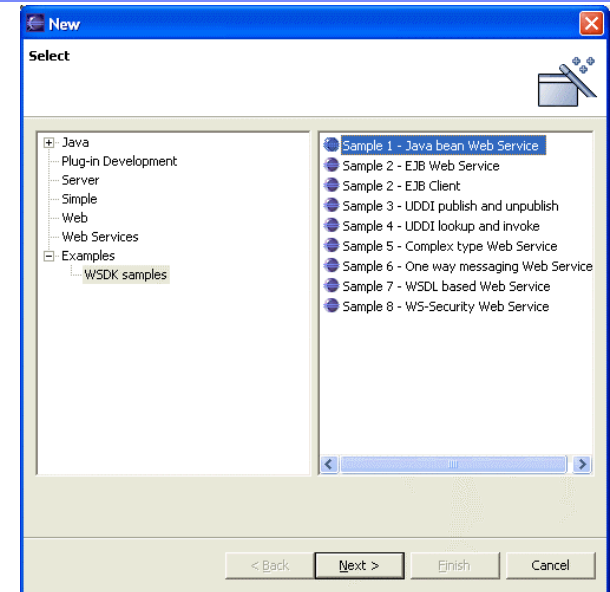
- The integrated kit for creating, discovering, invoking, and testing Web services – **FAST!**
 - Ready to go !
 - Provides the tools to create and test Web services in a tightly integrated package
 - Simplified Web application server, tools, documentation, and sample applications.
 - Interoperates with other vendors' implementations
 - specific samples for Microsoft® .NET clients.
 - Conforms to WS-I.org's Final Approved Basic Profile 1.0.
 - Easy to use !
 - Out-of-the-box solution for rapid building and testing of Web services.
 - Web services tools integrated with the Eclipse IDE
 - Comprehensive documentation, tutorials, and an online newsgroup.
 - Migration-ready !
 - Migration to enterprise-scale deployment with IBM® WebSphere® products.
 - Upward compatible from WSDK V5.0 and 5.01 versions

What you get with WSDK V5.1

- Application server
 - Embedded version of WebSphere Application Server – Express V5.0.2 with support for ORBs and Enterprise Java™Beans (EJBs)
- Standards support for :
 - SOAP 1.1, WSDL 1.1 UDDI 2.0, JAX-RPC 1.0, EJB 2.0, Enterprise Web services 1.0, WSDL4J, UDDI4J and WS-Security
- IBM WebSphere UDDI v2.0 registry
- Entry-level database with JDBC support
- IBM SDK for Java 2 Standard Edition (J2SE) version 1.3.1
- Eclipse plugins and samples showing how to :
 - expose JavaBeans, EJBs as Web services
 - enable browsing for Web services in UDDI registries
 - Create web services from WSDL defs
 - Publish / unpublish Web services to a UDDI registry
- Command line tools
- Information on how to use the Supply Chain example
- Comprehensive documentation including Web services concepts, developer tasks and reference materials

What's NEW in WSDK 5.1 ?

- ✓ Improved ease-of-use through Eclipse based GUI tools and Samples
 - Eclipse based GUI leverages existing Eclipse skills
 - Enhanced documentation
 - Quick start Samples
 - Alternative command line interface maintained
- ✓ Conformance to Final Approved WS-I Basic Profile 1.0
- ✓ Supply chain Management sample conforms to WS-I BP 1.0
 - Conforms to WS-I Usage Scenarios doc
 - Implements WS-I SCM sample architecture
- ✓ Full compatibility with WebSphere Studio family
 - Smooth transition from WSDK to WebSphere Studio Application Developer
- ✓ Full support for Split WSDL documents
- ✓ New WSDL2Client tool to create client-side only components
- ✓ Improved installation using Install Shield MultiPlatform V5.0.2



Installing WebSphere SDK For Web Services

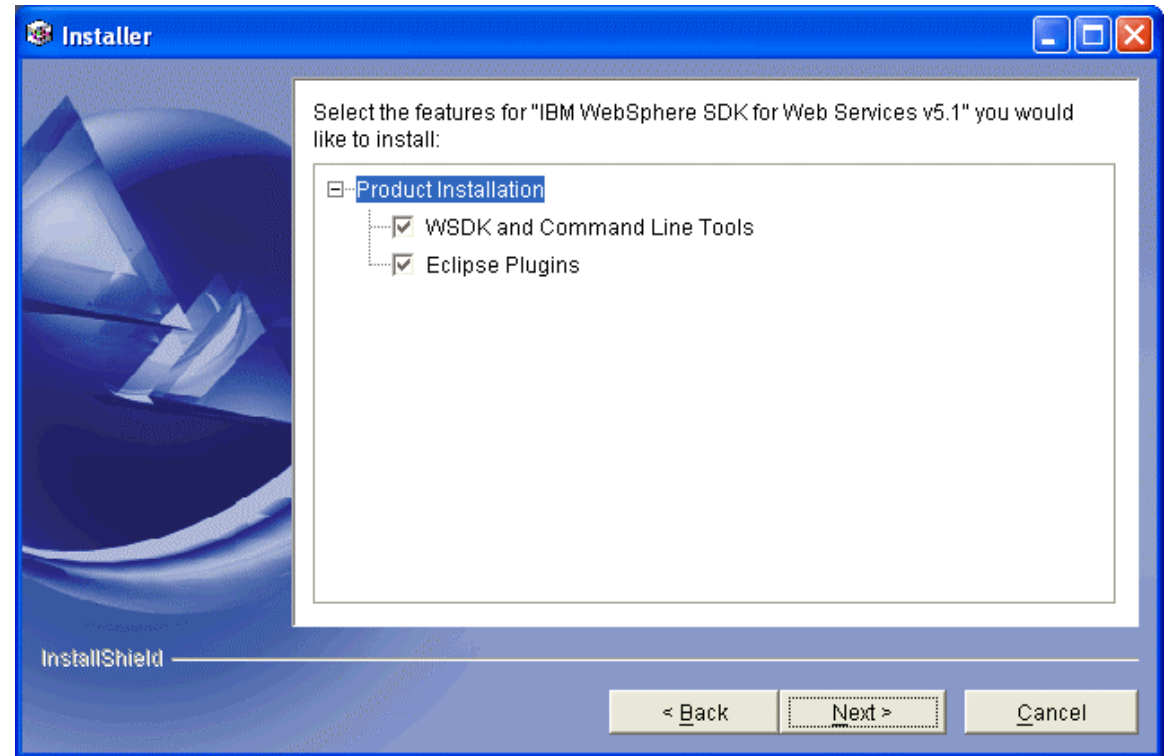
Click 1: What language?

Click 2: What to install?

Click 3: Where to Install ?

Click 4: Read Me ?

Click 5: You're done!



You can download the WSDK quickly (only 150MBs) and install in a few clicks

Getting Started with the WSDK (need new paths)

The WSDK provides comprehensive documentation, tutorials, and an online forum.

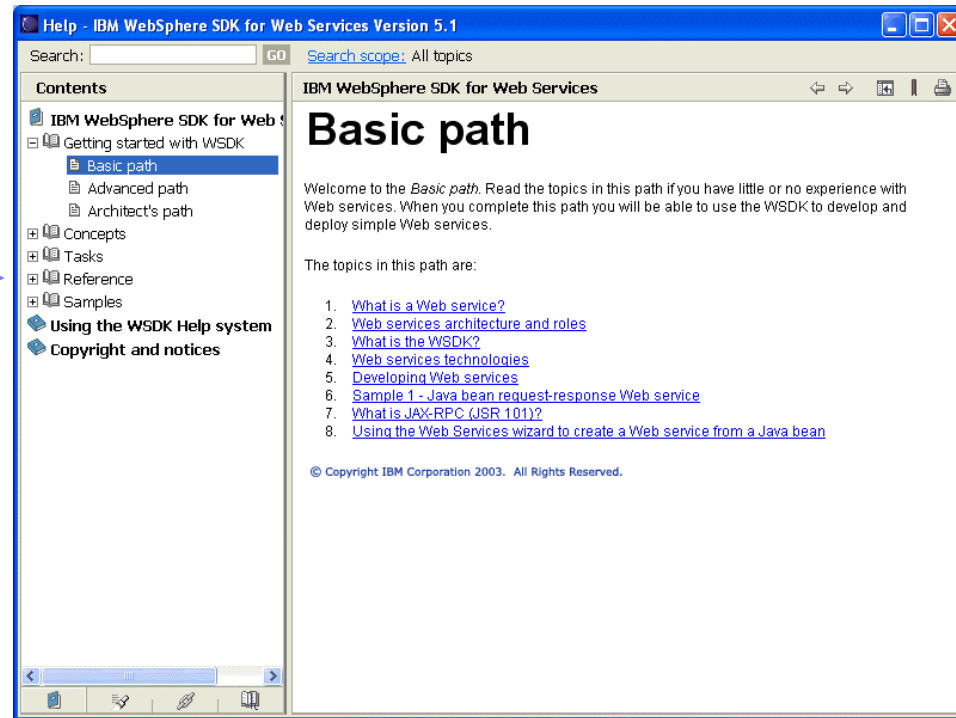


A reality check ... From an "experienced" developer, having never used the IBM WebSphere SDK for Web Services. "I was able to build, deploy and test a web service in just 10 minutes".

Getting Started with the WSDK (need new paths)

The WSDK provides comprehensive documentation, tutorials, and an online forum.

The Basic Path is for folks with little or no experience with Web services.



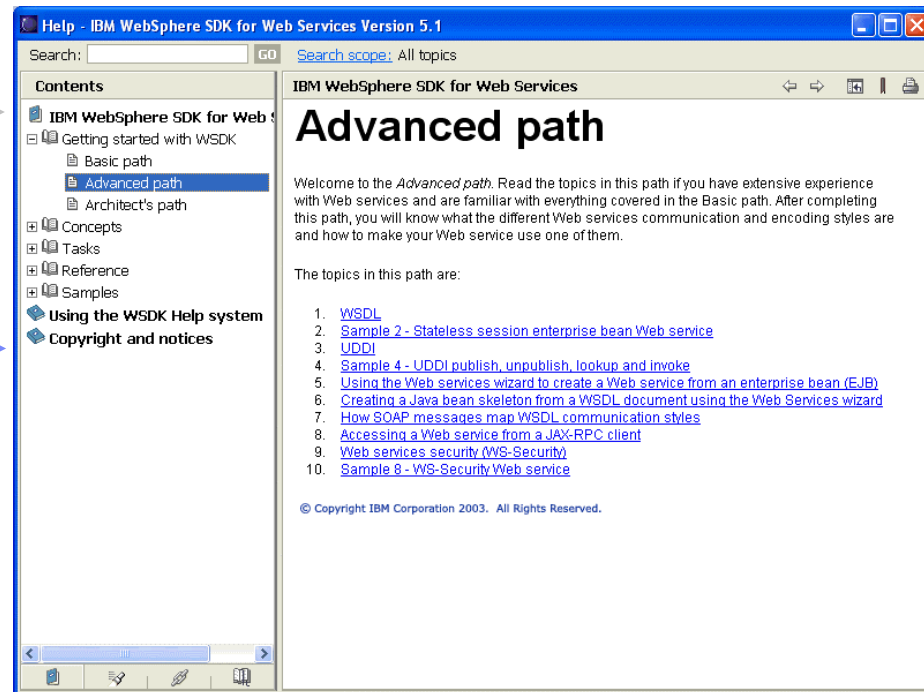
A reality check ... From an “experienced” developer, having never used the IBM WebSphere SDK for Web Services. “I was able to build, deploy and test a web service in just 10 minutes”.

Getting Started with the WSDK (need new paths)

The WSDK provides comprehensive documentation, tutorials, and an online forum.

The Basic Path is for folks with little or no experience with Web services.

The Advanced Path is for developers that have extensive experience with Web services.



A reality check ... From an "experienced" developer, having never used the IBM WebSphere SDK for Web Services. "I was able to build, deploy and test a web service in just 10 minutes".

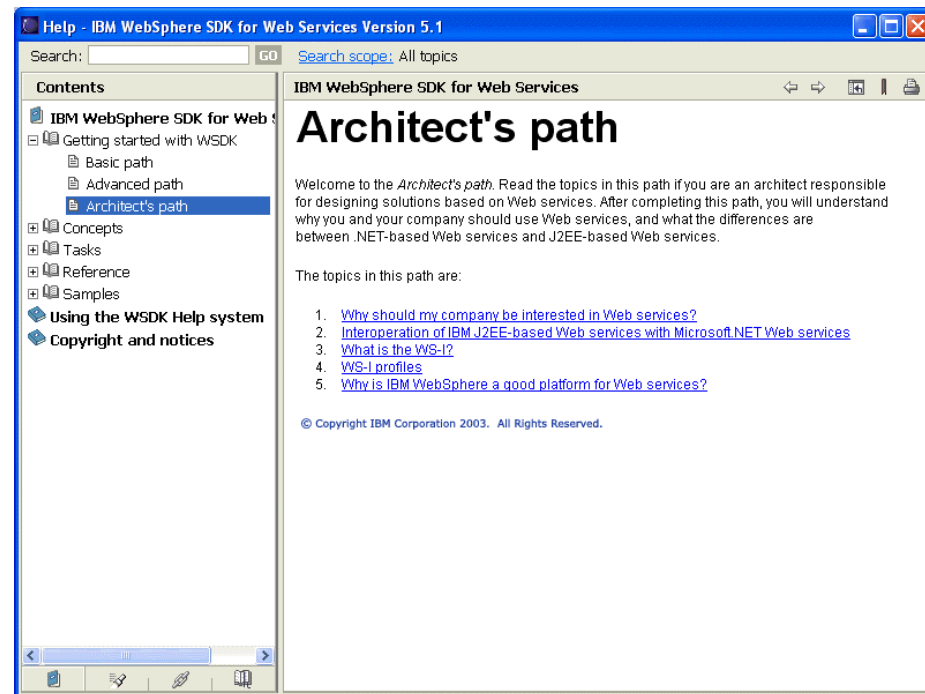
Getting Started with the WSDK (need new paths)

The WSDK provides comprehensive documentation, tutorials, and an online forum.

The Basic Path is for folks with little or no experience with Web services.

The Advanced Path is for developers that have extensive experience with Web services.

The Architect's path is for people responsible for designing solutions based on Web services.



A reality check ... From an "experienced" developer, having never used the IBM WebSphere SDK for Web Services. "I was able to build, deploy and test a web service in just 10 minutes".

Development and Usage Scenarios (generic to both)

- “Bottom up” enablement of existing components as Web services
 - Java class (JavaBean) or stateless session bean
 - “Java to WSDL” mapping
- “Top down” implementation of Web services from defined interfaces
 - “WSDL to Java” mapping
- Web services clients
 - J2SE application or J2EE application client
 - Static call via Service Endpoint Interface or DII-style dynamic call
 - “WSDL to Java” mapping
- Publication to UDDI (private or public registry)
- Different WSDL message styles: rpc/doc, literal/encoded

Developer Tools (command line tools)

Web service development (provider)

- Bean2WebService: enables existing JavaBean as Web service
- EJB2WebService: enables existing EJB as Web service
- WSDL2WebService: generates implementation template from existing WSDL, and creates application module

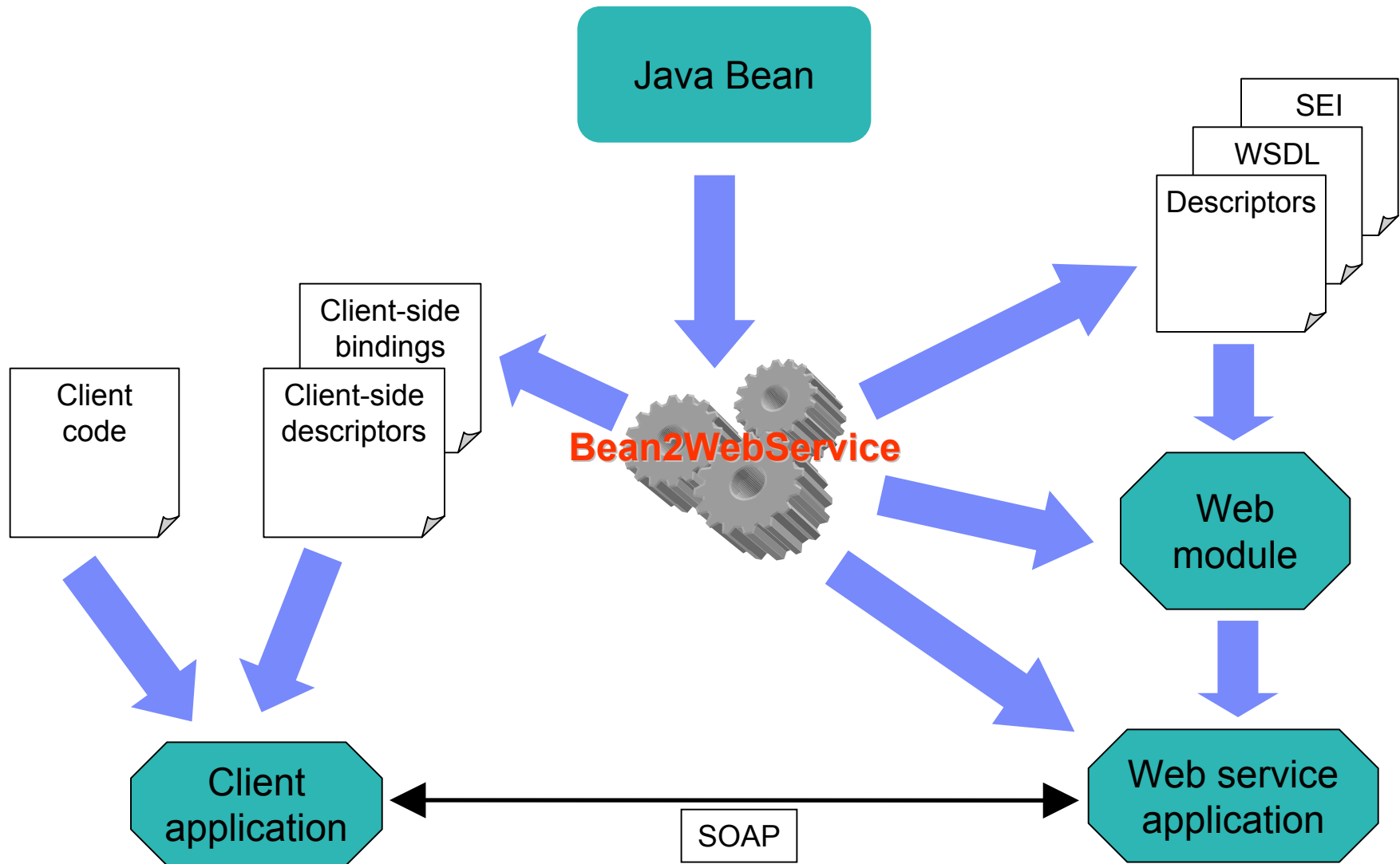
▪ Web service client development (requester)

- Use above tools or WSDL2Java tool to generate client bindings
- New tool, WSDL2Client: take WSDL and generate client code
- Use assembly tool (AAT) to build client components

▪ Other tasks

- UDDIPublish: publish to UDDI registry (also UDDIUnpublish)
- Administration: appserver, wsadmin
- Monitoring: tcpmon

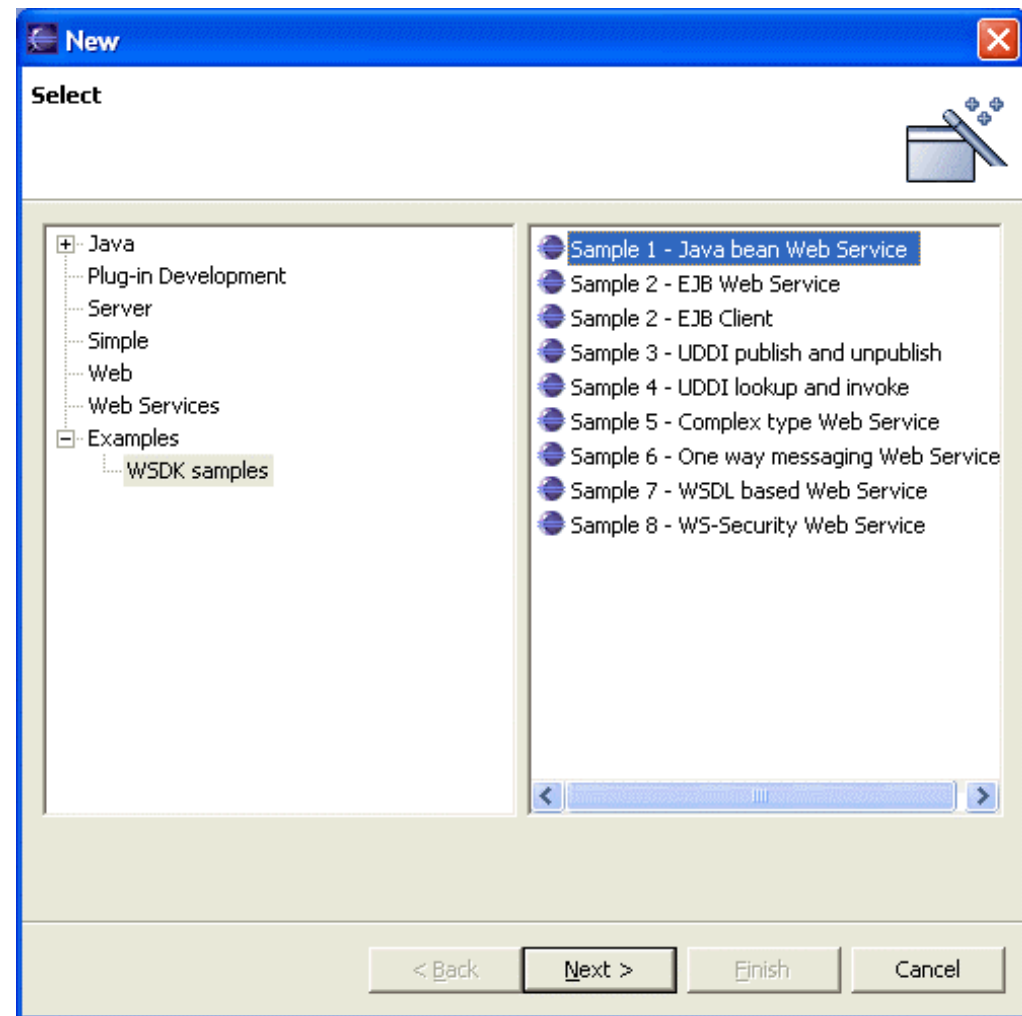
How to create a Web service (the easy way)



Interacting with Sample1 in Eclipse

Step 1:

Go to the File -> New -> Other wizard and select "Examples -> WSDK samples -> Sample1":



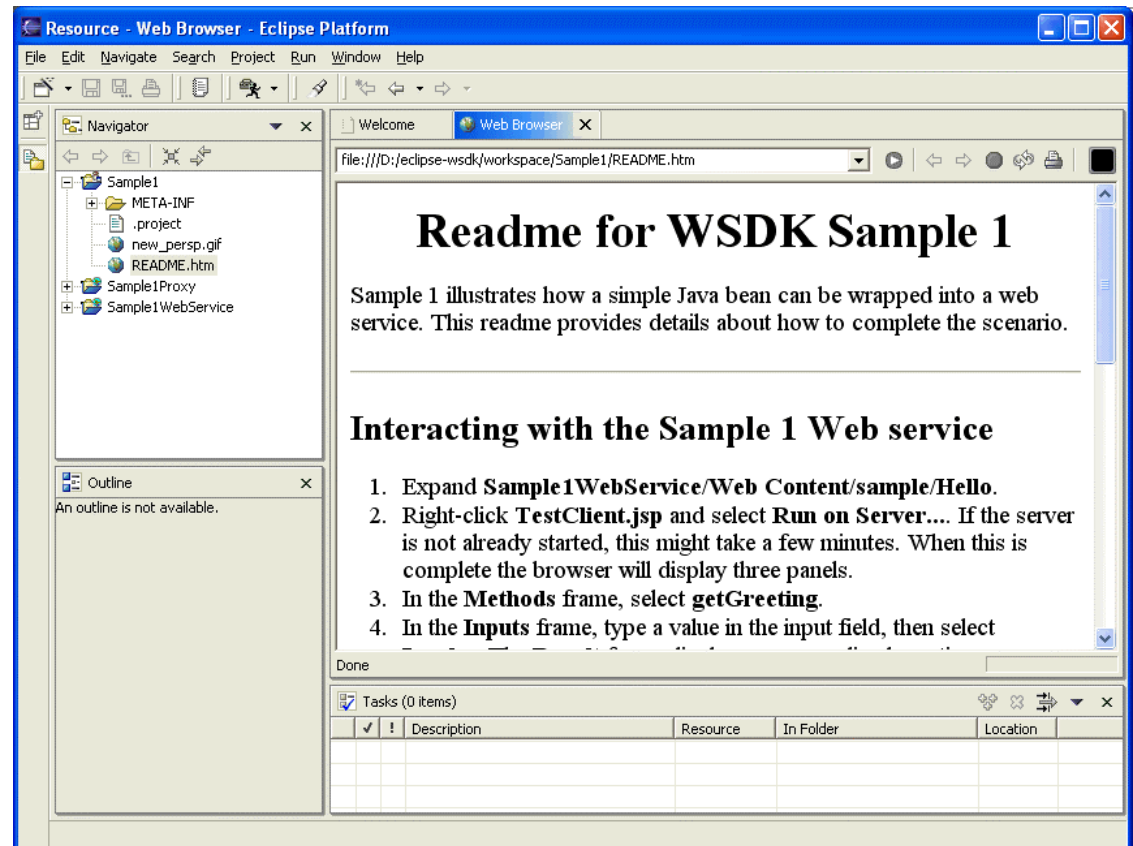
Interacting with Sample1 in Eclipse

Step 1:

Go to the File -> New -> Other wizard and select "Examples -> WSDK samples -> Sample1":

Step 2:

Next click, then Finish, and the sample is added to your workspace:



Interacting with Sample1 in Eclipse

Step 1:

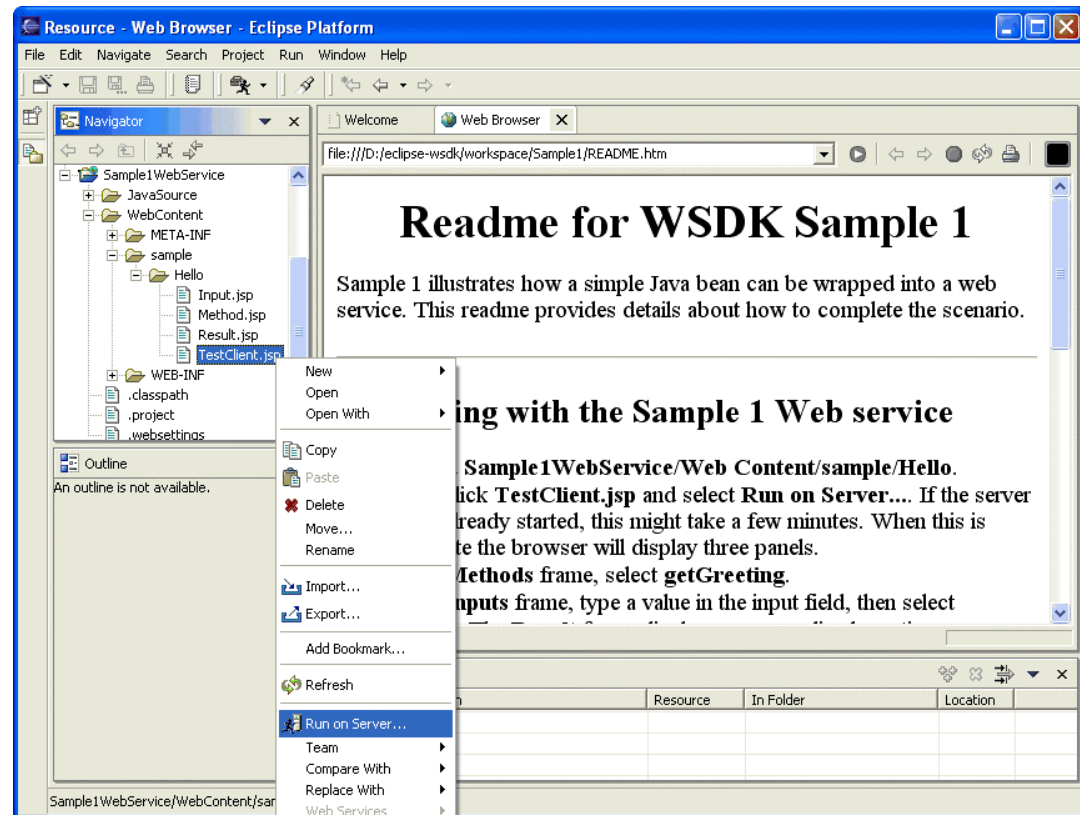
Go to the File -> New -> Other wizard and select "Examples -> WSDK samples -> Sample1":

Step 2:

Next click, then Finish, and the sample is added to your workspace:

Step 3:

Expand the folders and select "TestClient.jsp", right-click, then select "Run on Server..." from the popup menu:



Interacting with Sample1 in Eclipse

Step 1:

Go to the File -> New -> Other wizard and select "Examples -> WSDK samples -> Sample1":

Step 2:

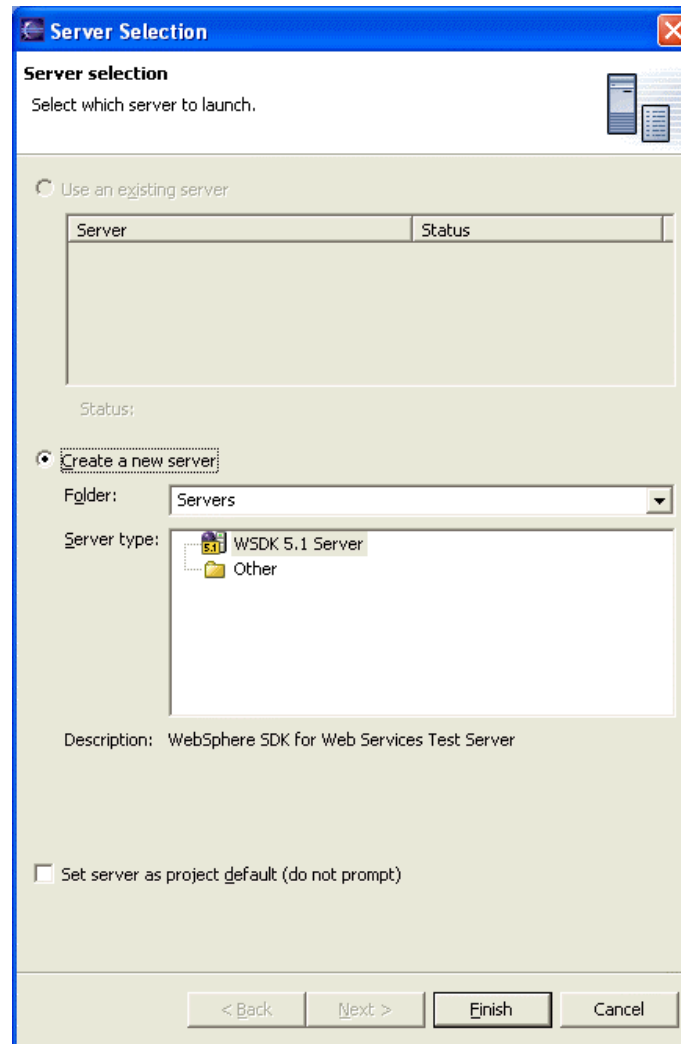
Next click, then Finish, and the sample is added to your workspace:

Step 3:

Expand the folders and select "TestClient.jsp", right-click, then select "Run on Server..." from the popup menu:

Step 4:

Choose the server: go with the defaults and just hit Finish:



Interacting with Sample1 in Eclipse

Step 1:

Go to the File -> New -> Other wizard and select "Examples -> WSDK samples -> Sample1":

Step 2:

Next click, then Finish, and the sample is added to your workspace:

Step 3:

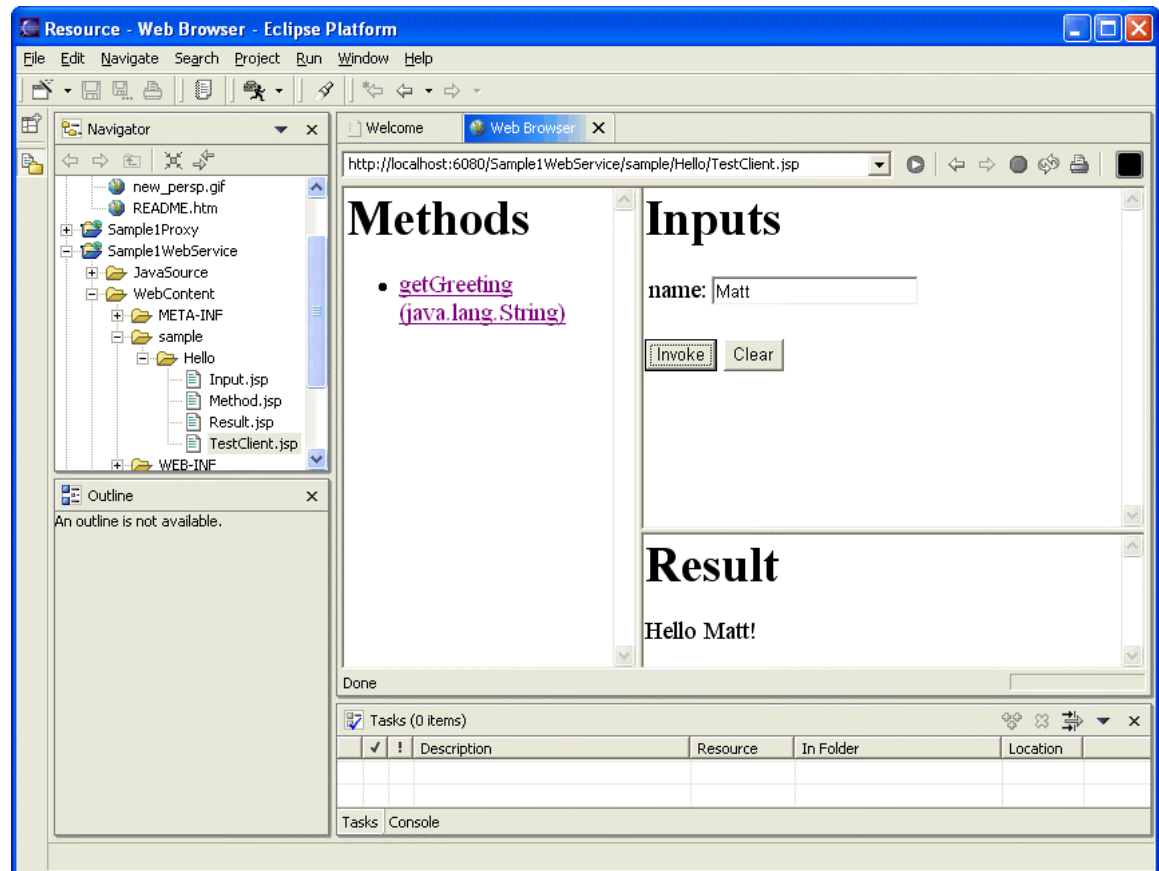
Expand the folders and select "TestClient.jsp", right-click, then select "Run on Server..." from the popup menu:

Step 4:

Choose the server: go with the defaults and just hit Finish:

Step 5:

The server is started (if it isn't already running), and the sample published. Select the "getGreeting" method on the left, enter some input, and hit Invoke to see the results:



Other Sample Applications

- “Bottom up” (Java to WSDL) and “top down” (WSDL to Java)
- JavaBean provider and stateless session bean provider
- Request/response and one-way messaging
- Simple types and complex types
- SOAP faults and headers
- Different client types:
 - J2SE client (JAX-RPC ServiceFactory)
 - J2EE application client (Java Naming and Directory Interface (JNDI) Service lookup)
 - DII client (UDDI lookup)
- UDDI: publish, unpublish, lookup
- Secure Web services (WS-Security)
- Instructions for building the samples using WSDK tools
- Demonstrate use of Split WSDL
- Provide a .NET Client, invokes a Java based Web service

WSDK and Emerging Technologies ToolKit

- ETTK is the successor to WSTK (as of the April 9, 2003 release).
- ETTK mission is to develop, integrate and enhance emerging technologies from IBM Research Labs and IBM Software Groups into a consolidated package, and to promote IBM leadership in these various emerging technologies. The first release of the ETTK concentrates on Autonomic, Grid and Web services technologies.
- Including Web services functionality in WSDK implies stability, industry standardization, and IBM commitment to deliver in upcoming product.
- ETTK provides additional functionality on top of WebSphere Application Server, WSDK or Tomcat. WSDK provides a complete and integrated stack, including an application server and the IBM Java SDK.
- ETTK is currently developing demos and implementations to align with future Autonomic and Grid solutions. In addition, the ETTK will provide new Common Event Formats/Web Service Management (Tivoli), On-Demand demos, and Web Services Business Processes. WSDK is delivering the core infrastructure for running these.

IBM WebSphere SDK for Web Services (WSDK)

- **WSDK home page and download:**
<http://www.ibm.com/developerworks/webservices/wsdk>
- **WSDK newsgroup:**
<news://news.software.ibm.com/ibm.software.websphere.webservices.sdk>
- **Speed-start Web services and WSDK tutorials:**
<http://www.ibm.com/developerworks/offers/ws-speed-start/>
- **WebSphere Version 5 Web Services Handbook:**
 - Go to <http://www.redbooks.ibm.com/>
and search for SG24-6891

© Copyright IBM Corporation 2003. All rights reserved.

IBM, the IBM logo, the e-business logo and other IBM products and services are trademarks or registered trademarks of the International Business Machines Corporation, in the United States, other countries or both. References in this publication to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates.

Product release dates and/or capabilities referenced in this publication may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries or both.

Microsoft, Windows, Windows NT and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries or both.

All other trademarks, company, products or service names may be trademarks, registered trademarks or service marks of others.