



IBM Software Group

IBM WebSphere Telecom Toolkit V6.2

Telecom Web services - Overview and features



@business on demand.

© 2007 IBM Corporation
Converted to video June 29, 2015

This presentation will provide an overview of the features of IBM WebSphere® Telecom Toolkit version 6.2 with emphasis on the Telecom Web Services feature of the toolkit.

Goals

- Introduce the Telecom Web services feature in WebSphere Telecom Toolkit
- Understand the usage of the tools available in the Telecom Web services feature



The goals of this presentation are to introduce the new development features of the Toolkit. These new and exciting features simplify various aspects of developing telecom client applications using telecom Web services and testing the applications using an integrated simulator. The simulator simulates a real telecommunication network for functional testing.

Agenda

- Overview
- Parlay x standards
- Tools
- Installation
- Wizards
- Telecom samples, snippets, cheat sheets
- Web services client simulator

The agenda for this presentation will start with an overview of the Telecom Web Services feature and talk about the Parlay X Standards and installation of the toolkit. The presentation will provide a brief understanding on the functionality and usage of the tools.

Overview

- Two installable features
 - ▶ Telecom Web services feature
 - ▶ IMS™ enablement feature
- Telecom Web services feature
 - ▶ Build and test telecom Web services client applications using Parlay X 2.1 API's
 - ▶ Features include
 - Wizards, cheat sheets, snippets, samples and a simulator.
- IMS enablement feature (not covered in this presentation)
 - ▶ Collection of tools to develop IMS applications



The WebSphere Telecom Toolkit has 2 features namely Telecom Web Services feature and the IMS Enablement feature. The Telecom Web Services feature provides a complete environment to develop and test Telecom client applications using Parlay X 2.1 APIs. The feature includes wizards, cheat sheets, snippets, samples and a simulator that emulates a real telecommunication network. Each of these features is explained in greater detail later in the presentation. The IMS Enablement feature provides a collection of tools for developing other IP Multimedia Subsystem applications. This presentation covers only the Telecom Web Services feature of the toolkit.

Installation

- Extension to Rational® Application Developer V7.0
- Available on Windows® and Linux® operating systems
- Requires WebSphere Application Server 6.1.0.3 and above
- Installed as a feature of WebSphere Telecom Toolkit V6.2
- Getting Started guide includes installation and removal procedures

The Toolkit is built as an extension to Rational Application Developer version 7.0. It is available for Windows XP and Linux operating systems that Rational Application Developer 7.0 supports. The toolkit provides an Installshield installer. The integrated simulator in the Telecom Web Service feature requires the WebSphere Application Server 6.1.0.3 fix pack. Before installing the Toolkit, Rational Application Developer should be closed. Rational Application Developer ships WebSphere Application Server version 6.1.0.2 as the server runtime. The getting started guide shipped with the toolkit covers the instructions to install and uninstall the toolkit for Windows and Linux operating systems.

Parlay X standards

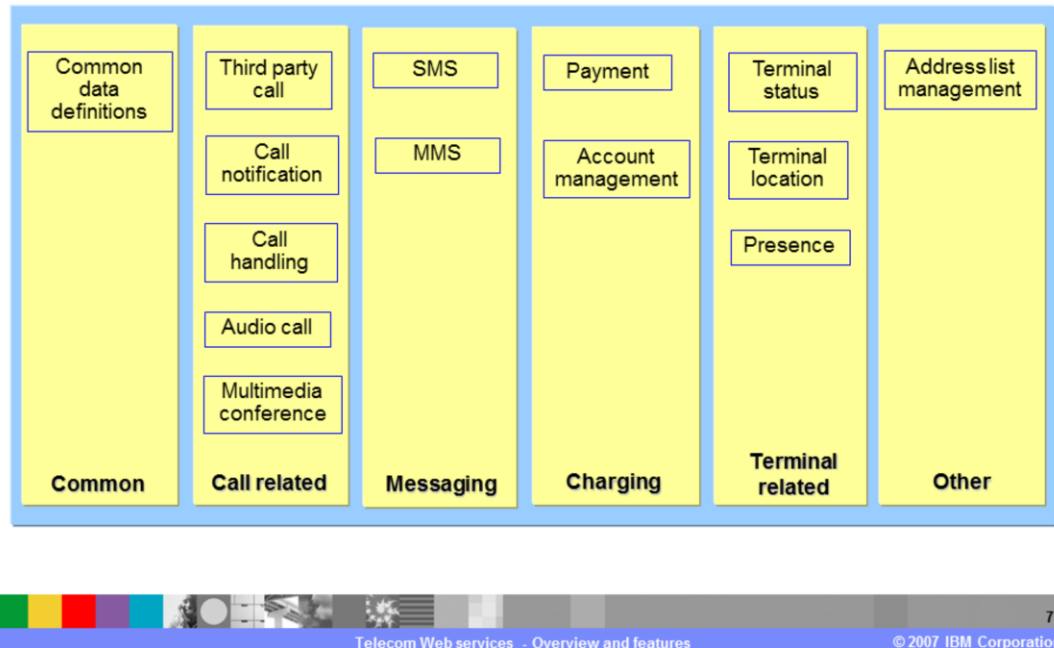
- Set of telecommunication Web services and related APIs
- Defined jointly by European Telecommunications Standards Institute (ETSI), Parlay, and the Third Generation Partnership Program (3GPP).
- Specification has fourteen parts

Available at: <http://parlay.org/en/specifications/pxws.asp>.



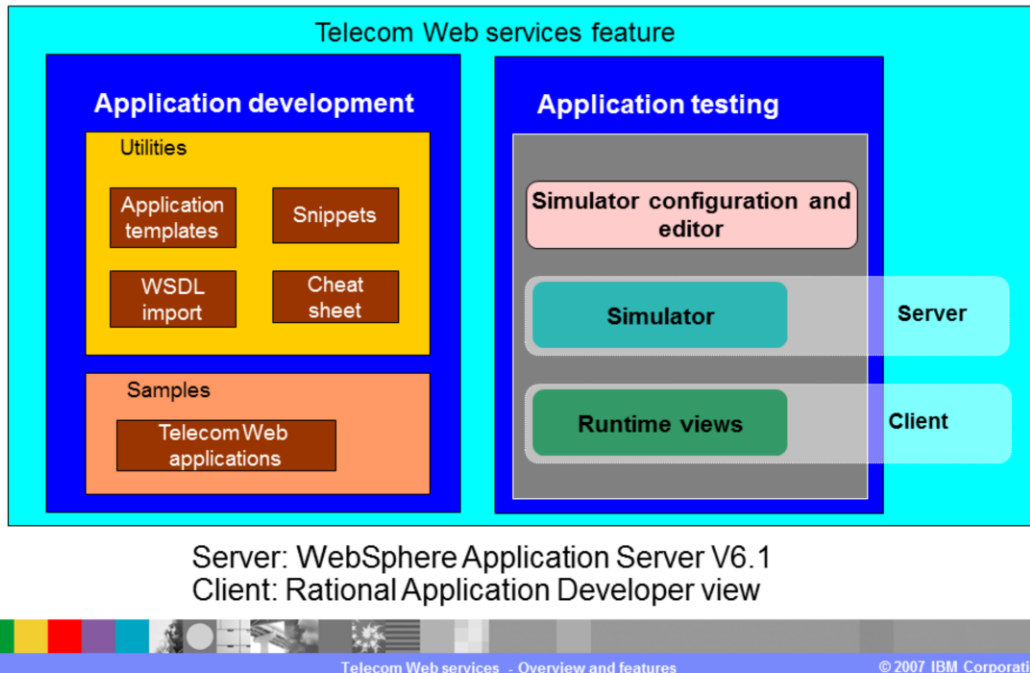
The Toolkit is built based on Parlay X 2.1 open standards. The Parlay X is a set of Telecommunication Web Services and API supporting these services. These Web services allow an application to use the capabilities of a telecommunication network. The Parlay X specification standard has been jointly developed by European Telecommunications Standards Institute and 3rd Generation Partnership Project. It is made up of 14 parts as described in the next chart. The specifications and the WSDLs can be downloaded from the Parlay Web site at the mentioned Web address.

Categorized Parlay X 2.1 Services/API



The 14 Parlay X services are categorized into 6 major categories. These categories are summarized as shown in this slide. The resources in the “Common” category are required for all categories. “Call Related Services” category includes services that abstract the telecommunication call related capabilities. “Messaging Services” category includes services capable of sending and receiving Text and Multimedia messages. “Charging Services” category provides services for charging related functionality such as Payment and Account Management. “Terminal Related Services” relate to a terminal or device. The “Other” category lists any service that does not fall into the other categories.

Big picture



This slide provides the complete set of tools provided by the Telecom Web Services feature. To provide clarity the tools are categorized into application development tools and application testing tools. The application development tools include the telecom application templates, snippets, WSDL import wizard, cheat sheets and telecom Web samples. The Application Testing tools include the simulator configuration Editor, the simulator and the runtime Views.

Tools

▪ **Wizards**

- ▶ Import Wizard to import Parlay X 2.1 WSDLs
- ▶ Wizard to create new Telecom simulator configuration file

▪ **Cheat sheets:** Guided steps to create a Telecom Web application

▪ **Snippets:** Insert code to call these Web services:

- ▶ Message, Terminal Location, Account Management, Call Notification, Payment, Third Party Call, Audio Call, Group Management, Group Member Management, Presence and Wireless Access Protocol (WAP) Push.



The various tools available in the Telecom Web Services feature include:

- Two wizards which are:
 - The Parlay X 2.1 WSDL import wizard used to import Parlay 2.1 WSDLs into a existing project.
 - The telecom simulator configuration wizard used to create a new custom simulator configuration file.
- The telecom cheat sheets provide guided steps to create a new application using the telecom Web application template sample.
- The telecom snippets are used to insert predefined working code in a Java class or Java Server Page. There are snippets to call various Parlay x Web services that include:
 - Short Message Service
 - Multimedia Message Service
 - Terminal Location
 - Terminal Status
 - Accounting
 - Payment
 - Notification Administration
 - Third Party Call
 - Audio call
 - Group Management
 - Presence
 - Wireless Access Protocol push

Tools

▪ Samples

- ▶ Telecom samples are sample Web service client applications
- ▶ Telecom Web application template is a sample template to develop new Web service client applications

▪ Integrated Web services client simulator

- ▶ To test the telecom Web service client applications.
- ▶ Configured for testing by the simulator configuration
- ▶ Simulator configuration editor used to edit the simulator configuration
- ▶ Simulator runtime views display simulator runtime data



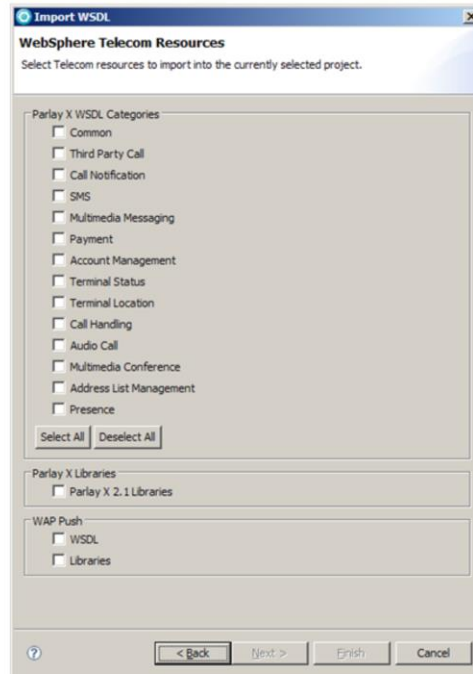
Two samples are shipped with the Toolkit. The Telecom samples built using the toolkit that can be deployed and tested on the Web services client simulator. The Telecom Web Application Template sample provides a template Web project to develop new applications. The template when imported into the Rational Application Developer workspace creates Web and Enterprise Application projects. The Web project contains all the required parlay x API jars on its classpath.

The telecom Web services feature includes a Web Services Client Simulator, a simulator configuration editor and various simulator runtime views. The Web Services Client Simulator emulates a parlay x gateway and provides a test suite to test user developed Parlay x Web applications without the need of a real network. The simulator uses a configuration file to configure its test data. It is an xml file with a wss file extension that has static configuration of the Simulator's environment that dictates how services will behave.

The Toolkit ships a default simulator configuration along with the Telecom Samples. The user can create a custom simulator configuration using the Telecom Simulator Configuration wizard as explained in the earlier slide. The simulator configuration editor is used to edit a user created custom simulator configuration. The simulator runtime views display the simulator runtime data while the Simulator is in operation.

Wizards

- Import wizard
- To import Parlay X 2.1 WSDLs into the selected project
 - ▶ Select a Project -> Import -> Telecom -> Parlay X 2.1 WSDL to launch the wizard
- Can be imported into any kind of a project.



11

Telecom Web services - Overview and features

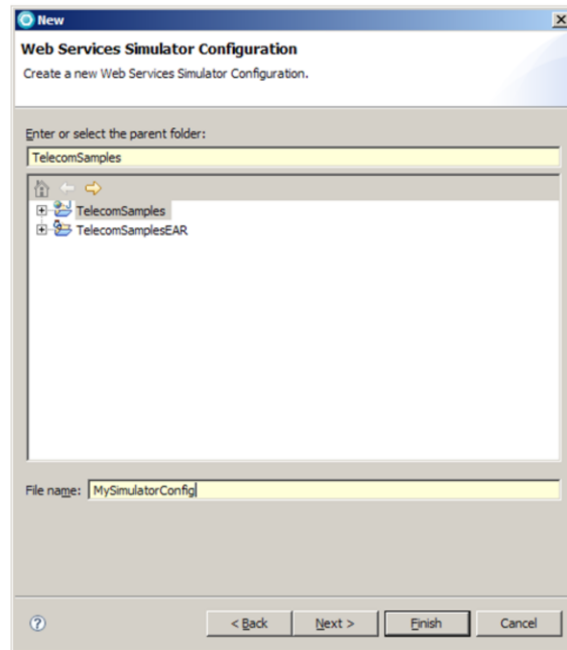
© 2007 IBM Corporation

The Import WSDL wizard is used to import Parlay X 2.1 WSDLs and libraries into an existing project. The Parlay X WSDL Categories section provides controls to import the WSDLs related to all the 14 Parlay X 2.1 services. The Parlay X Libraries sections provides an option to import the Parlay X 2.1 API's that are packaged in the "parlayx21.jar." The WSDL can be imported into a Java or a non Java project. Similarly WAP Push resources can also be imported.

Wizards

Telecom simulator configuration wizard

- ▶ Wizard to create new simulator configuration file
- ▶ Provides nested wizards to create new accounts and devices for the accounts



12

Telecom Web services - Overview and features

© 2007 IBM Corporation

The Telecom Simulator Configuration Wizard is used to create a new custom simulator configuration for use with the simulator. The simulator uses a default configuration which is an xml file with the wss extension to populate the simulation data with various service policies for the telecom services. The simulator can be configured to use the custom configuration instead of the default configuration.

The Configuration Wizard uses all the default service policy values while creating the new configuration which can be edited later using the configuration editor. The configuration wizard contains nested wizards to create new accounts and new devices for each account. The figure on this slide shows the first page of the configuration wizard. You need to enter a new configuration file name in the File name text entry. Clicking Next on the first page gives you an option to create new accounts and devices.

The Telecom Simulator Configuration wizard can be launched using the menu options File -> Other -> Telecom category.

Samples

- Samples (Help->Samples Gallery)
 - ▶ Telecom samples
 - Demonstrates use of Parlay X 2.1 API
 - Includes examples for various Telecom services
 - ▶ Telecom Web application template
 - A dynamic J2EE Web project template with dependencies to parlay X 2.1 Web services
 - Provides a structured pattern for developing Web services client parlay x 2.1 applications.

The Telecom Web Services feature ships 2 samples in the samples gallery.

The samples shipped by the toolkit are under Help -> Samples Gallery -> Technology -> Telecom Web Services.

The Telecom Samples demonstrate the use of Parlay X 2.1 API in writing applications.

The Telecom Web Application Template sample is a dynamic J2EE Web project. The project contains the basic dependencies and Web application structure needed to begin developing Parlay X 2.1 Web service client applications. Both the samples are shipped as J2EE enterprise applications that can be deployed to WebSphere Application Server version 6.1.

Snippets in Rational Application Developer

- A collection of reusable programmable objects.
- A mechanism to demonstrate best practices
- Ease development with ready made code
- Snippets in Rational Application Developer show up in the Snippets view
- Each entry in Snippet view is a catalogue of snippets

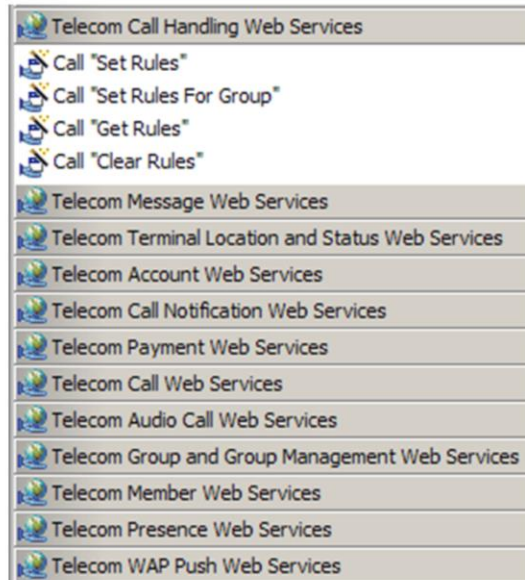
Snippets in Rational Application Developer are collection of reusable programmable objects to aid application development.

The programmable objects could be a chunk of Java Code, JSP Tags or JavaScript. The programmable objects are provided in Rational Application Developer to ease the development of applications and advice developers using best practices. Developers can insert these programmable objects in their application by populating any variables values required by the snippet.

The snippets in application developer show up in the Snippets view that is available in the J2EE perspective. Each entry in the snippet view represents a catalogue of snippets. Clicking on any snippet entry in the snippet view opens a drawer of snippets. Each snippet in the snippets view launches a wizard if the reusable programmable object expects any variable values.

Telecom snippets

- Call Parlay X 2.1 Web services
- Figure shows all Telecom snippet categories
- Clicking each catalogue opens a drawer of Snippets.
- Double clicking a snippet launches snippet wizard



15

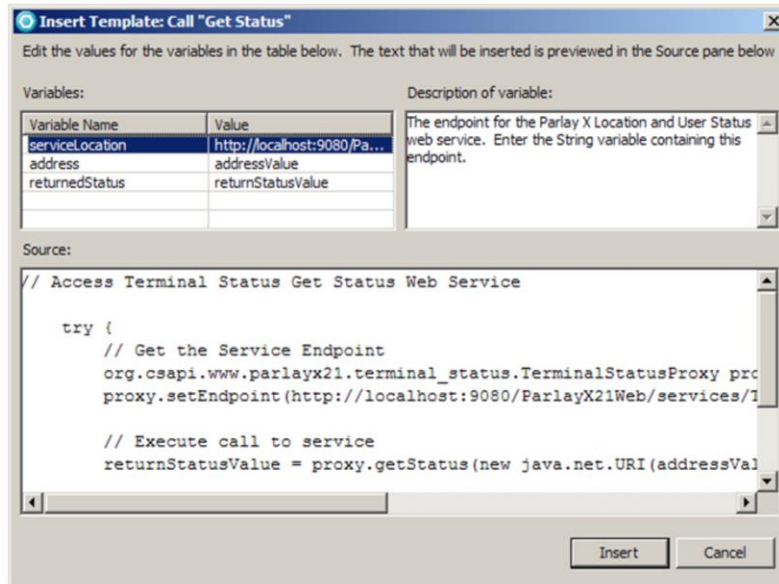
Telecom Web services - Overview and features

© 2007 IBM Corporation

The Telecom Web Services feature extends the snippets view to provide several telecom snippets. They provide reusable Java code to call various Parlay X 2.1 Web services.

The figure in this slide shows all the telecom snippets provided by the Telecom Web Services feature. Clicking on any of the telecom snippet entry opens a drawer of telecom snippets as shown in the figure for "Telecom Call Handling Web Services". Double clicking on any of the telecom snippet launches a snippet wizard to customize the code with variable values.

Telecom snippet wizard



16

Here's a snapshot of the snippet wizard launched. The snippet wizard is used to insert variable values for the code inserted by the snippet into a Java class or Java server page. The variables table shows the mandatory and optional variables that need to be set to complete the snippet code. The Description area shows the description of the selected variable in the variables table and if it is optional or not. The Source area previews the resulting code that inserted into the method.

Telecom snippets

- Telecom snippet catalogues
 - ▶ Telecom message services
 - ▶ Telecom terminal location and status Web services
 - ▶ Telecom account Web services
 - ▶ Telecom notification administration Web services
 - ▶ Telecom payment Web services
 - ▶ Telecom call Web services
 - ▶ Telecom audio call Web services
 - ▶ Telecom group and group management Web services
 - ▶ Telecom member Web services
 - ▶ Telecom call handling Web services
 - ▶ Telecom address list management Web services
 - ▶ Telecom wireless access push Web services

All the telecom snippets use parlay X 2.1 API and provide code to call various parlay X 2.1 Web services. Each of the snippet categories contains a set of snippets.

The Telecom Message Services category provides snippets to send and get Multimedia Message Service and Short Message Service.

The Telecom Terminal Location and Status Web Services category provides snippets to get terminal location and start, end notification of location information for a device or group of devices.

The Telecom Account Web Services category provides snippets to get the Account details for an Account (For example Balance, Expiry Date, History and so on). The Telecom Notification Administration Web Services category provides snippets to create get and end Call, SMS and MMS, Location and terminal status notifications.

The Telecom Payment Web Services category provides snippets for Payment services that include Charging, Refund, Regular and Volume Reservations.

The Telecom Call Web Services category provides snippets to make Third Party Calls and getting the call information.

The Telecom Audio Call Web Services category provides snippets to play text, audio and voice messages, get message status and end a message.

The Telecom Group and Group Management Web Services category provides snippets to create delete and query groups, group members and attributes of group members.

The Telecom Member Web Services category provides snippets to add delete and query member attributes.

The Telecom Presence Web Services category provides snippets to get current presence information, subscribe to the presence information of a presentity and create notifications for subscribe and presence services for presentities.

The Telecom Call Handling Web Services category provides snippets to set get and clear rules for a group of members.

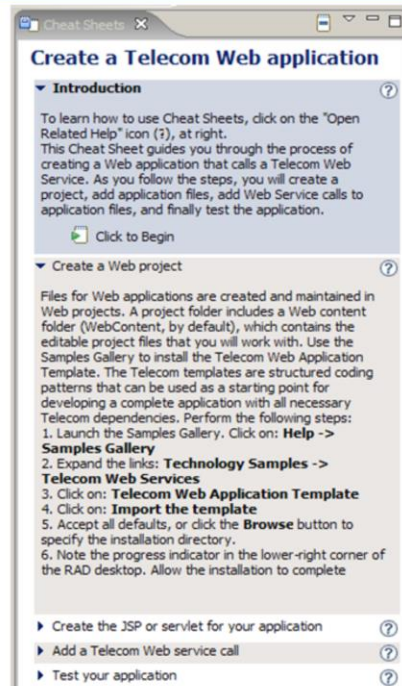
The Telecom Address List Management Web Services category provides snippets to add remove and query member attributes of a group of members.

The Telecom Wireless Access Push Web Services category provides snippets to send a WAP Push request and query the delivery status of the request.

Note: The word Terminal and Device are used interchangeably and mean the same.

Telecom cheat sheets

- Cheat sheets in Rational Application Developer provide a guided path to perform a task.
- Telecom Web application cheat sheet
 - ▶ Guides the process of creating an Web application that calls a Telecom Web service.



18

Telecom Web services - Overview and features

© 2007 IBM Corporation

Cheat sheets in Rational Application Developer guide you through some of the application development processes. Each cheat sheet is designed to help you complete some task, and it lists the sequence of steps required to help you achieve that goal. The telecom Web services feature provides telecom cheat sheets that are designed to guide you through the process of developing and testing a Telecom Web services application using manual steps. In Rational Application Developer the cheat sheet selection dialog is launched by selecting Cheat Sheets menu item under Help.

In the cheat sheet selection dialog select “Create a Telecom Web Application” from the Telecom folder and click OK. The figure shows a screen capture of the telecom cheat sheet. The telecom cheat sheets provide four steps to help develop and test a Parlay x 2.1 telecom client application. The four steps are:

1. Create a Web Project - guides you through the process of loading the Telecom Web Application Template.
2. Create the JSP or Servlet for your application - guides you through the process of creating a client that could be a JSP or a servlet to call a telecom Web service.
3. Add a Telecom Web Service call - guides the process of adding Web service call in the JSP or servlet client using the Telecom Snippets.
4. Test your application - guides the process of testing user client application using the Web services client simulator running on the integrated WebSphere Application Server version 6.1 in Rational Application Developer.

Web service client simulator overview

- Emulates a Parlay X gateway without the need of communicating with actual gateway
- Provides complete test environment to test Parlay X 2.1 applications
- Simulates the network for accounts, devices, groups and Parlay X 2.1 Web services.
- Configurable using a simulator configuration file
- Deployed as J2EE enterprise application on WebSphere Application Server 6.1

The Toolkit ships a Web services client simulator to test Parlay x 2.1 client applications. It provides a self-container environment to debug and test parlay x 2.1 applications. The simulator emulates a Parlay X gateway without the need of an actual gateway. It simulates the telecom network for Accounts, Devices, Groups of users and the other Parlay x Web services. It is configurable using the simulator configuration which is an XML file. It is deployed as an enterprise application in the integrated WebSphere Application Server version 6.1 test environment of application developer.

Web service client simulator overview (cont..)

- Provides runtime views displaying the session data of devices configured in the simulator
 - ▶ Runtime views available – Activity, Call, Device, Group, Map and Presence
 - ▶ Data in some views can be modified using the editable fields to allow rapid application testing with different input values.
 - ▶ Runtime views are connected to the simulator when simulator is deployed and simulator client is running.
 - ▶ Click the Connect button in runtime views to connect to the simulator if the views do not connect automatically

The Toolkit has several runtime views that display the current session data of various devices and the services invoked by these devices in the simulator.

The runtime views include the Activity view, Call view, Device view, Group view, Map view and the Presence view. The data in some of the runtime views can be modified to allow rapid testing of the application with different input values. The runtime views are connected and display the current simulator data only when the simulator is running.

If runtime views are not connected automatically when the simulator is running, you can connect the views to the simulator using the “Connect” button in each view.

Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM IMS Rational WebSphere

Rational is a trademark of International Business Machines Corporation and Rational Software Corporation in the United States, Other Countries, or both.

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

J2EE, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprocessing in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2007. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.