

# Extension Mobility API Sample Application

A blue square icon containing a white stylized globe with latitude and longitude lines.

## Disclaimer

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS DOCUMENT ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS DOCUMENT ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Table of Contents

**Overview..... 3**

**Source Packages .....3**

**Build Environment.....3**

**Installation.....3**

**User Interface.....3**



# Overview

This is a sample application written to show how to use the Extension Mobility API as documented in the 7.0(1) Cisco Unified Communication Manager XML Developer Guide available at the Cisco Developer Community web site.

## Source Packages

The source files are broken into several packages. The DTD files and this overview.html file are in the root of the src folder. These will show up in NetBean <default packages> section. The EMTool package contains the core java files for the project. The helpers package contains the response helper classes used by the EMDtd class. The loginrequest, loginresponse, queryrequest and queryresponse packages are autogenerated by the xjc tool.

## Build Environment

1. JDK 1.6
2. NetBeans 6.5
3. Cisco Unified Communications Manager 7.0(1)
4. Tool was installed and executed on a Windows XP Service Pack 2 desktop

## Installation

1. Unzip the EMTool.zip package to the desired directory
2. Start the tool by executing the EMTool.jar file found in the \EMTool\dist directory.
3. If desired, open JavaDocs by executing the index.html file found in the \EMTool\dist\javadoc directory.

## User Interface

The user interface is a simple Swing interface created using the Netbeans Form Editor. Here is an overall view of the User Interface.



The User Information section contains information used to identify the user, device and profile that are used with the Extension Mobility feature.

The Cisco Unified Communications Manager section contains information used to identify the location of the EMService and also to provide a username and password for a user that has the capability to perform Extension Mobility functions.

The Login, Logout, Query Device and Query Users buttons perform the four basic tasks.

The Login function has an optional parameter which allows an auto logout feature after a specified number of minutes.

The Implementation section allows the choice of using either the DTD/Jaxb implementation or the Raw Text implementation.

The Options window has check boxes that allow you to show or hide the results window, which has the basic results from each function. The Console window displays all output to System.out and System.err.

Note that when the Console is closed, System.out and System.err are restored to their default values.