



USSD Gateway Simulator

User Guide v1.2

Version control

Version	Date	Author	Comment
R0	2009/08/03	Alejandro Leib	Initial Version
V1.1	2010/09/24	Alejandro Leib	Phone without buttons
V1.2	2010/10/07	Alejandro Leib	Phone registry config

Notice

This document contains proprietary and confidential material of LeibICT. Any unauthorized reproduction, use, or disclosure of this material, or any part thereof, is strictly prohibited. This document is solely for the use of LeibICT employees and authorized LeibICT customers.

The material furnished in this document is believed to be accurate and reliable. However, no responsibility is assumed by LeibICT for the use of this material. LeibICT reserves the right to make changes to the material at any time and without notice.

Copyright 2007 LeibICT.

LeibICT

Montevideo, Uruguay Tel: +598 2 614 11 93 Fax: +598 2 203 66 54

<http://www.leibict.com>

Warranty

LeibICT Solutions keep a technical support department with the only purpose of providing efficient and reliable service.

All LeibICT products are warranted against defects in material and workmanship. The period of coverage and other warranty details are specified in the LeibICT terms and conditions warranty. In no event shall LeibICT be liable for incidental or consequential damages in connection with, or arising from use of any LeibICT product.

Table of Contents

Introduction.....	4
Simulation	5
Running the Simulator	6
Running the Application.....	8
Simulating a Service	10
HTTP Interface Simulation.....	11

Introduction

LeibICT UDK (USSD Developer Kit) is the first comprehensive kit for documentation, samples and simulators that enables any Java or C++ programmer -either on Linux (32 and 64bit variants) or Windows- to develop a USSD Service within minutes.

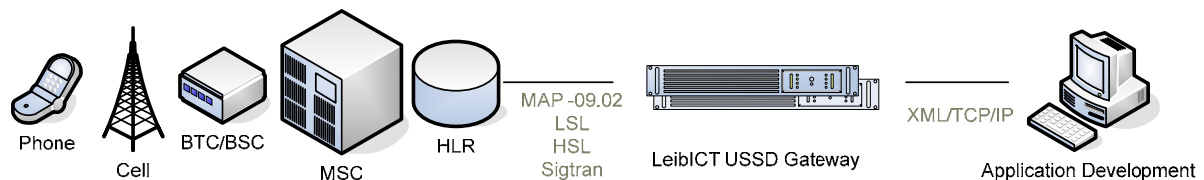
The kit includes **LeibICT Phone Simulator**, a revolutionary tool based on the LeibICT USSD GW Simulator, that enables the developers to test their application within a 'phone skin', very useful for presentations too!.

The Simulators are applications for near-to-live testing of your USSD services without need of access to a real UMTS/GSM network and they behave as a real HLR with MAP interface, so that your services can send and receive messages, however nothing will get delivered anywhere as all the responses are only made-up by the Simulator. The Simulators also support multiple simultaneous USSD transactions.

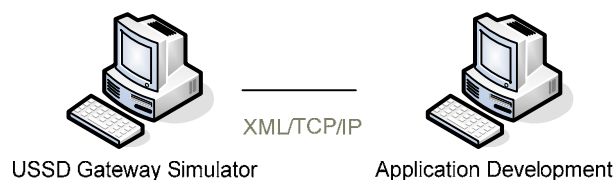
Switching from the Phone Simulator or the USSD GW Simulator to our USSD S-Gateway links is an effortless task, making time to market even shorter.

Simulation

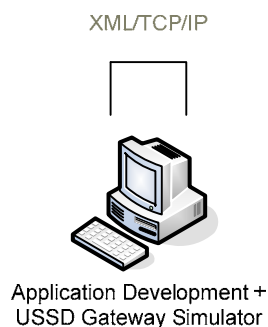
The normal operation of a USSD Gateway involves the utilization of a real UMTS/GSM network:



The simulator simply removes the need of all the UMTS/GSM network components:



Also enabling a loopback configuration if you run the the simulator module in the development environment:



The simulators are Windows Applications, so you will be able to develop only Windows Applications running on the same machine of the Simulator although you may use VMWare and other tools to avoid this.

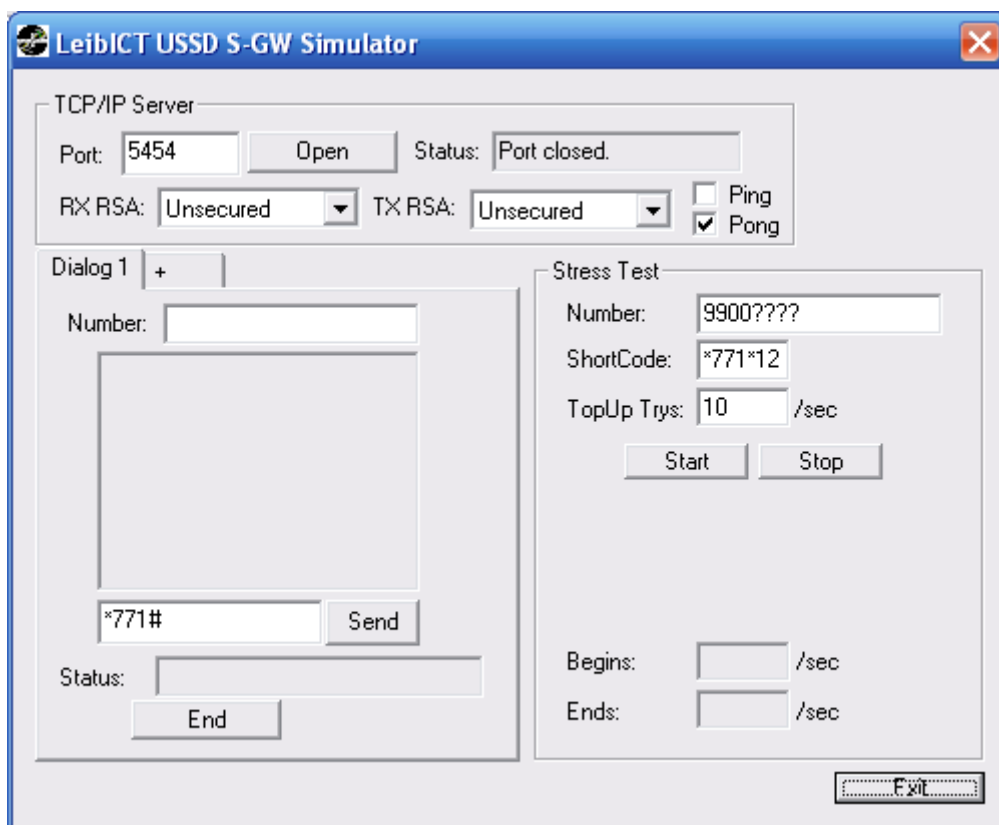
If you like to develop Linux, Solaris, or Mac based applications, a VMWare or another machine is required.

Running the Simulator

To run the Windows based Simulator just double-click on the exe file:

Important note: there are two simulators provided on the UDK and you must only use one at a time, both simulators cannot be used in parallel or in other cooperative way.

The multi session USSD GW Simulator:



To start a simulation, first open the TCP/IP port: click the “Open” button.

The Status will change to “Port open. Waiting for App.”

The Phone skinned Simulator:



The TCP/IP port is automatically opened at 5404.

To send and clear digits just press the “real” buttons.

If you need to modify this Simulator MSISDN or TCP/IP port just click on your Windows

- 1) Start -> Run.. -> regedit (the Windows Registry Editor application will open)
- 2) Go to HKEY_LOCAL_MACHINE\SOFTWARE\LeibICT\Phone
- 3) Modify the keys oMSISDN and/or port
- 4) Close the regedit and restart the simulator.

Running the Application

In the samples directory you will find several sample applications, we will use the Java based USSDTest:

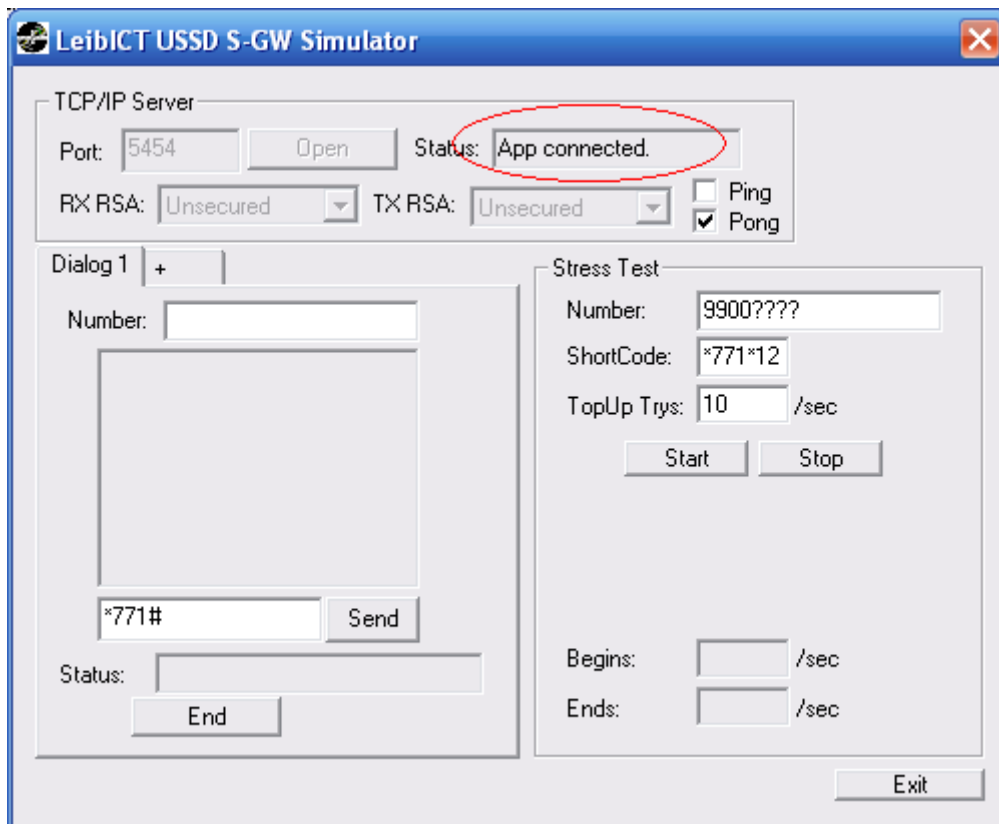
```
java -jar USSDTest.jar
```

The result screen is:

LeibICT - USSD Java Native Interface Windows Library v1.0b build 1002

```
logDir          = c:\logs\  
logDebug        = 0  
log             = 1  
logFileName     = ussdjni.log  
logTrafficFileName = ussdjniTraffic.log  
logTraffic      = 0  
logErrorsFileName = ussdjniErrors.log  
JavaVM JNI Load...  
JavaVM JNI Load OK  
Creating USSD JNI Object...  
Creating USSD JNI Object OK  
Xpath version 1001  
Creating Class Reference...  
Creating Class Reference OK  
Registering Callbacks...  
Registering Callbacks OK  
Initializing Socket...  
Initializing Socket OK  
Connecting localhost:5454...  
Connected to localhost:5454  
Securing connection...  
Securing connection OK  
cb_connected
```

Note that the cb_connected callback is fired and the simulator Status is changed to “App Connected”:



The screenshot shows the LeibICT USSD S-GW Simulator window. The title bar reads "LeibICT USSD S-GW Simulator". The interface is divided into several sections:

- TCP/IP Server:** Contains a "Port" field with the value "5454", an "Open" button, and a "Status" field displaying "App connected." (circled in red). Below these are "RX RSA" and "TX RSA" dropdown menus, both set to "Unsecured". To the right are checkboxes for "Ping" (unchecked) and "Pong" (checked).
- Dialog 1:** Includes a "+" button, a "Number:" text box, a large empty text area, and a "Send" button. Below the text area is a status field showing "*771#" and an "End" button.
- Stress Test:** Contains a "Number:" text box with "9900????", a "ShortCode:" text box with "*771*12", and a "TopUp Trys:" text box with "10 /sec". There are "Start" and "Stop" buttons. Below these are "Begins:" and "Ends:" text boxes, each followed by a "/sec" label.
- Exit:** A button located at the bottom right of the window.

Simulating a Service

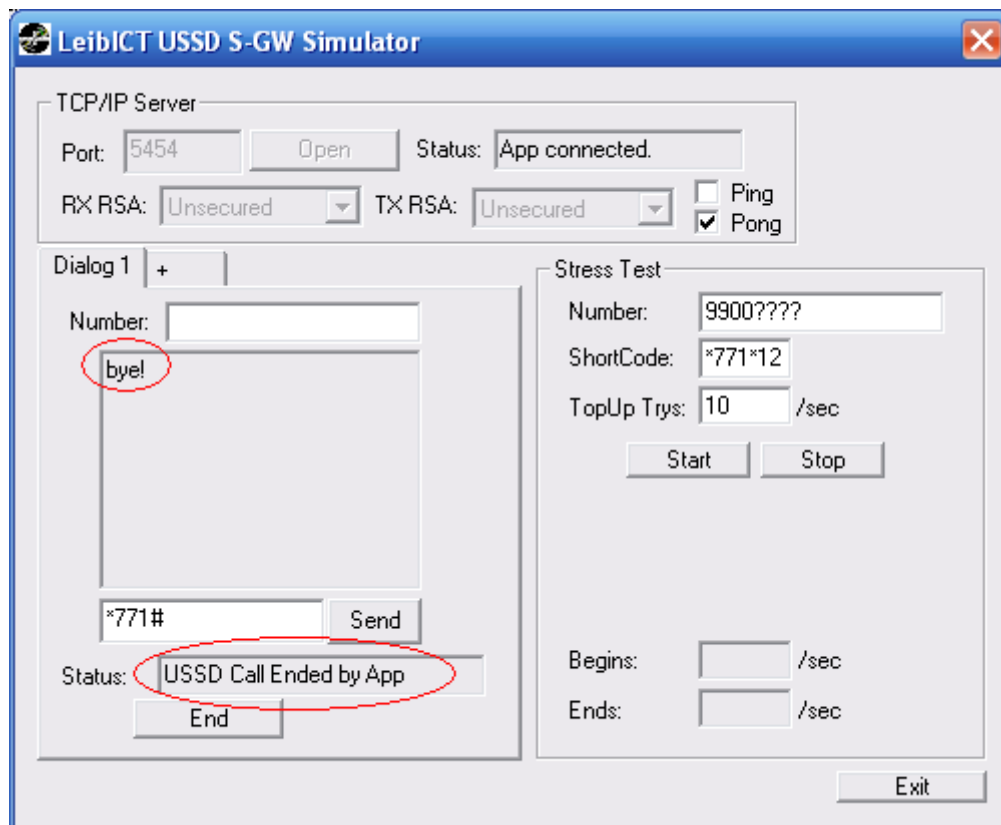
The Simulator is ready and the Application is ready, its time to simulate a USSD Service transaction.

To start a service just click “Send” with the text on any USSD short code you need:

The application will receive that:

1 *771#

And in case of the USSDTest.jar I will end the USSD transaction with a “bye” message:



HTTP Interface Simulation

- 1) Start the normal Simulation: double click on ussdsgwSimulator.exe
- 2) Click the Open button on the Simulator
(unblock it from the firewall, it will open the port 5404)
- 3) Execute **uhgw.exe** for 1st time
(it will create Windows registry keys)
- 4) Close **uhgw**
- 5) Execute (start->run) "regedit"
- 6) Got to "HKEY_LOCAL_MACHINE\SOFTWARE\LeibICT\UHGW"
(HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\LeibICT on 64 bits windows)
- 7) Configure the values "logDir" and "httpurl" properly
- 8) Start **uhgw.exe** and check the logs.
- 9) Start simulating as normal.