Inter-process communication

URemote Application

|  |  |  |
| --- | --- | --- |
| Author | Date | Description |
| Cyril Leroux | 24/10/2012 | Document creation |
|  | 28/08/2013 | Updated formatting |
|  | 10/05/2014 | Messages Refactoring |
|  | 09/01/2015 | Updated request codes |

Table of Contents

[I. Protocol Buffer 2](#_Toc408563424)

[II. The physical layer 2](#_Toc408563425)

[III. Data structure 2](#_Toc408563426)

[1. Overview 2](#_Toc408563427)

[2. Request Types 3](#_Toc408563428)

[3. Request Codes 3](#_Toc408563429)

[4. Implemented combinations 5](#_Toc408563430)

[Type Keyboard 5](#_Toc408563431)

[Type AI 6](#_Toc408563432)

[Type Volume 6](#_Toc408563433)

[Type App 6](#_Toc408563434)

[IV. Handling keyboard messages in Windows (server side) 6](#_Toc408563435)

[V. Next features 7](#_Toc408563436)

# Protocol Buffer

To communicate with other systems URemote implements Google Protocol Buffers.

“Protocol Buffers are a way of encoding structured data in an efficient yet extensible format. Google uses Protocol Buffers for almost all of its internal RPC protocols and file formats.”

See <https://code.google.com/p/protobuf/> for further informations.

# The physical layer

The client application is designed to run on a mobile device such as a phone or a tablet.

Most of the devices have at least Bluetooth and Internet connections (over Wi-Fi or 3G).

For other data transmission supports we will have to use a hardware relay that receives commands over Bluetooth, Wi-Fi or 3G and resends them over IR, RF, Zeegbee or else.

# Data structure

## Overview

Protocol buffers is based on messages. URemote has been developed to work with two main messages: Request and Response.

Request is composed by

* A type of request
* Two codes (primary, and extra)
* A security token
* An integer parameter (used to pass extra integer data)
* A string parameter (used to pass extra string data)

Response is composed by

* The request data (type and primary code)
* A return code (RC\_ERROR or RC\_SUCCESS)
* A string message
* An integer value
* A FileInfo message (another message type that represents a file or directory)

## Request Types

|  |  |  |
| --- | --- | --- |
| Label | Value | Description |
| SIMPLE | **0** | For basic uncategorized commands (such as ping). |
| EXPLORER | **1** | Browse, open and launch remote file system. |
| KEYBOARD | **2** | Remote keyboard. |
| AI | **3** | AI related commands (mute). |
| VOLUME | **4** | Volume related commands. |
| APP | **5** | To launch and control applications remotely. |

## Request Codes

| Label | Value | Description |
| --- | --- | --- |
| NONE | **0** |  |
| DEFINE | **1** | The value will be either a string or an integer (define in request extra params) |
| STATUS | **2** |  |
| ON | **3** |  |
| OFF | **4** |  |
| PING | **5** |  |
| TEST | **6** |  |
| KILL\_SERVER | **7** |  |
| SHUTDOWN | **8** |  |
| SWITCH\_WINDOW | **9** |  |
| LOCK | **10** |  |
| MUTE | **11** |  |
| SAY | **12** |  |
| DPAD\_UP | **13** | Direction arrow |
| DPAD\_DOWN | **14** | Direction arrow |
| DPAD\_LEFT | **15** | Direction arrow |
| DPAD\_RIGHT | **16** | Direction arrow |
| Values 20-25: Media and file browser keys | | |
| MEDIA\_PLAY\_PAUSE | **20** |  |
| MEDIA\_STOP | **21** |  |
| MEDIA\_PREVIOUS | **22** |  |
| MEDIA\_NEXT | **23** |  |
| MEDIA\_FF | **24** |  |
| MEDIA\_REWIND | **25** |  |
| Values 26-29: File browser keys | | |
| QUERY\_ROOTS | **26** | Query the available roots of the remote device. |
| QUERY\_CHILDREN | **27** |  |
| OPEN\_SERVER\_SIDE | **28** |  |
| OPEN\_CLIENT\_SIDE | **29** |  |
| Values 30-39: Modifier keys | | |
| KEYCODE\_CTRL | **30** |  |
| KEYCODE\_SHIFT | **31** |  |
| KEYCODE\_ALT\_LEFT | **32** |  |
| KEYCODE\_ALT\_RIGHT | **33** |  |
| KEYCODE\_WINDOWS | **34** |  |
| Values 40-49: Keyboard special keys | | |
| KEYCODE\_ENTER | **40** |  |
| KEYCODE\_ESCAPE | **41** |  |
| KEYCODE\_DELETE | **42** |  |
| KEYCODE\_SPACE | **43** |  |
| KEYCODE\_BACKSPACE | **44** |  |
| KEYCODE\_TAB | **45** |  |
| Values 500-599: Keyboard basic keys | | |
| KEYCODE\_0 | **500** |  |
| … |  |  |
| KEYCODE\_9 | **509** |  |
| KEYCODE\_A | **510** |  |
| … |  |  |
| KEYCODE\_Z | **535** |  |
| KEYCODE\_F1 | **550** |  |
| … |  |  |
| KEYCODE\_F12 | **561** |  |
| Values 600-700: Keyboard special characters keys | | |
| KEYCODE\_EQUALS | **600** | = |
| KEYCODE\_MINUS | **601** | - |
| KEYCODE\_PLUS | **602** | + |
| KEYCODE\_STAR | **603** | \* |
| KEYCODE\_SLASH | **604** | / |
| KEYCODE\_BACKSLASH | **605** | \ |
| KEYCODE\_UNDERSCORE | **606** | \_ |
| KEYCODE\_PIPE | **607** | | |
| KEYCODE\_COMMA | **608** | , |
| KEYCODE\_PERIODE | **609** | . |
| KEYCODE\_COLON | **610** | : |
| KEYCODE\_SEMICOLON | **611** | ; |
| KEYCODE\_AT | **612** | @ |
| KEYCODE\_APOSTROPHE | **613** | ‘ |
| KEYCODE\_LEFT\_PARENT | **650** | ( |
| KEYCODE\_RIGHT\_PARENT | **651** | ) |
| KEYCODE\_LEFT\_BRACKET | **652** | [ |
| KEYCODE\_RIGHT\_BRACKET | **653** | ] |
| KEYCODE\_LEFT\_CURLY\_BRACKET | **654** | { |
| KEYCODE\_RIGHT\_CURLY\_BRACKET | **655** | } |
| KEYCODE\_LEFT\_ANGLE\_BRACKET | **656** | < |
| KEYCODE\_RIGHT\_ANGLE\_BRACKET | **657** | > |

## Implemented combinations

### Type Keyboard

* Combine with all KEYCODE\_\* codes ;

### Type AI

### Type Volume

|  |  |  |
| --- | --- | --- |
| Code | Extra | Result |
| DPAD\_UP | - | Increase volume |
| DPAD\_DOWN | - | Decrease volume |
| DEFINE | An integer value | To specify the wanted volume |

### Type App

|  |  |  |
| --- | --- | --- |
| Code | Extra | Result |
| QUERY\_ROOTS | - | Return the available remote applications |
| ON | The app id | Start the specified application |
| OF | The app id | Kill the specified application if running |

# Handling keyboard messages in Windows (server side)

void Keyboard::CtrlEnter()  
{

keybd\_event(VK\_CONTROL, 0, 0, 0);  
keybd\_event(VK\_RETURN, 0, 0, 0);  
keybd\_event(VK\_RETURN, 0, KEYEVENTF\_KEYUP, 0);  
keybd\_event(VK\_CONTROL, 0, KEYEVENTF\_KEYUP, 0);

}

void Keyboard::AltF4()  
{

keybd\_event(VK\_LMENU, 0, 0, 0);  
keybd\_event(VK\_F4, 0, 0, 0);  
keybd\_event(VK\_F4, 0, KEYEVENTF\_KEYUP, 0);  
keybd\_event(VK\_LMENU, 0, KEYEVENTF\_KEYUP, 0);

}

**void** Keyboard**::**SendKeyboardInput(WORD \_keyCode)  
{

KEYBDINPUT kbInput;  
 kbInput.wVk **=** \_keyCode;  
 kbInput.wScan **=** 0;  
 kbInput.dwFlags **=** 0;  
 kbInput.time **=** 0;  
 kbInput.dwExtraInfo **=** (ULONG\_PTR) GetMessageExtraInfo();

INPUT input;  
 input.type **=** INPUT\_KEYBOARD;  
 input.ki **=** kbInput;

SendInput(1, **&**input, **sizeof**(INPUT));

}

**void** Keyboard**::**SendKeyboardInput(WORD \_code, WORD \_extraCode)  
{

cout **<<** "in Keyboard::SendKeyboardInput(WORD \_code, WORD \_extraCode)" **<<** endl;  
 cout **<<** "-- code : " **<<** \_code **<<** endl;  
 cout **<<** "-- extra code : " **<<** \_extraCode **<<** endl;

INPUT combined\_inputs[] **=**

{

{ INPUT\_KEYBOARD, MAKELONG(\_extraCode, 0) },  
 { INPUT\_KEYBOARD, MAKELONG(\_code, MapVirtualKey(VK\_SPACE, 0)) },  
 { INPUT\_KEYBOARD, MAKELONG(\_code, MapVirtualKey(VK\_SPACE, 0)), KEYEVENTF\_KEYUP },  
 { INPUT\_KEYBOARD, MAKELONG(\_extraCode, 0), KEYEVENTF\_KEYUP }

};

SendInput(ARRAYSIZE(combined\_inputs), combined\_inputs, **sizeof** INPUT);

}

# Next features

* Network discovery (search for an URemote server in the current subnetwork)
  + <http://developer.android.com/training/connect-devices-wirelessly/nsd.html>
  + <http://stackoverflow.com/questions/3717332/android-find-upnp-devices-from-network>
* Custom logger in the C++ server
* Anymote protocol implementation
  + <https://code.google.com/p/anymote-protocol/>
  + <https://code.google.com/p/google-tv-pairing-protocol/>
  + <https://code.google.com/p/google-tv-remote/>
  + <https://play.google.com/store/apps/details?id=com.google.android.apps.tvremote>
  + <https://www.kickstarter.com/projects/1635386542/anymote-home-your-phone-the-ultimate-universal-rem>
* Enhance device to device communication (IR, RF, Wifi, Bluetooth)
  + <http://www.lirc.org/>
  + <http://www.cooking-hacks.com/documentation/tutorials/control-hvac-infrared-devices-from-the-internet-with-ir-remote>