CounterPath Corporation

Suite 300, Bentall One Centre
505 Burrard Street Box 95
Vancouver BC V7X 1M3
Canada V6B1R8

Telephone: +1.604.320.3344
www.counterpath.com

Bria 3 *for Windows* API

Developer Guide

CounterPath Corporation
Suite 300, One Bentall Centre
505 Burrard Street, Box 95
Vancouver, BC V7X 1M3
Tel: 604.320.3344
sales@counterpath.com   www.counterpath.com

This document contains information proprietary to CounterPath Corporation, and shall not be used for engineering, design, procurement, or manufacture, in whole or in part, without the consent of CounterPath Corporation.

CounterPath and the logo are trademarks of CounterPath Corporation.

The content of this publication is intended to demonstrate typical uses and capabilities of the CounterPath Bria 3 *for Windows* API from CounterPath Corporation. Users of this material must determine for themselves whether the information contained herein applies to a particular IP-based networking system. CounterPath makes no warranty regarding the content of this document, including—but not limited to—implied warranties of fitness for any particular purpose. In no case will CounterPath or persons involved in the production of this documented material be liable for any incidental, indirect or otherwise consequential damage or loss that may result after the use of this publication.

Windows, Windows Vista, and Win32 are registered trademarks of Microsoft Corporation in the United States and other countries.

This manual corresponds to Bria 3.1 *for Windows*.

Rev 3

The CRM API

The CRM (customer relationship management) API provides functions that let another application log information about calls made using Bria. This API can be used along with the CLI API (next page) to manage phone calls.

For example, your application may have a web page that provides information about customers for your employees to phone.

- The employee clicks on the customer phone number, which invokes a function from the CLI API to place the phone call via Bria.
- As soon as the call is placed, a back-end component of your application (which is subscribing to Bria events via the CRM API) will receive information about the call from Bria.
- The employee will manage the call using Bria: the employee can put the call on hold, transfer the call, mute the call, and so on.
- When the call ends, the back-end component will receive notification that the call has ended.

Receiving Call Events from Bria

To receive call events from Bria:

1. Implement an interface to listen for the WM_COPYDATA message that is passed in the Msg parameter of the Win32 SendMessage() function.

2. To listen to Bria:

   - Register for the Request Listener message by calling RegisterWindowMessage(), passing in a pointer to the string “CP_REQUEST_LISTENER”.
   - Register for the Listener Confirmed message by calling RegisterWindowMessage(), passing in a pointer to the string “CP_LISTENER_CONFIRMED”.
   - Implement an interface to listen for messages that are handled by WndProc. Listen for both Win32.WM_COPYDATA and CP_LISTENER_CONFIRMED.
   - Call PostMessage()with the CP_REQUEST_LISTENER message as the Msg parameter and your application’s hWnd handle as the hWnd parameter.

   Your application should receive a CP_LISTENER_CONFIRMED WndProc message from Bria confirming that the listener is now the destination for Bria call messages.

3. When an event occurs on Bria, Bria calls SendMessage(), passing:

   - The hWnd (originally passed in PostMessage()).
   - WM_COPYDATA in the Msg parameter.
   - A pointer to the APIMESSAGE structure in the lParam’s COPYDATASTRUCT. This APISTUCTURE contains all the information for the call event. See the next page.

APIMESSAGE

The structure of the APIMESSAGE is as follows (in C#):

```
public struct APIMESSAGE
        {
           public int messageType;
           [MarshalAs(UnmanagedType.ByValTStr, SizeConst = 1024)]
           public string localUri;
           [MarshalAs(UnmanagedType.ByValTStr, SizeConst = 1024)]
           public string remoteUri;
           [MarshalAs(UnmanagedType.ByValTStr, SizeConst = 1024)]
           public string message;
        }
```

- messageType is an enum for either outgoing call, incoming call, and call ended messages. (More messages will be implemented in the future.)

```
      public enum MessageTypeEnum
            {
               callOut = 0,
               callIn = 1,
               callEnd = 2,
            }
```

- localUri is the local account URI which Bria used to place or receive the call.
- remoteUri is the remote party’s URI.
- Message has content only for messageType CallEnd. It contains the duration of the call in seconds in this field. For an unanswered call, the duration is 0 seconds.

The CLI API

The CLI (command line interface) API lets you execute Bria from a command line.

From the Command Line

Currently, the API supports the following actions:

- Bria3.exe -action=call?to=<remoteuri>
- Bria3.exe -action=videocall?to=<remoteuri>
- Bria3.exe -action=IM?to=<remoteuri>

- Bria3.exe -action=add?contact=<contact URI>
- Bria3.exe -action=add?group=<name>
- Bria3.exe -action=delete?contact=<contact URI>
- Bria3.exe -action=delete?group=<name>
- Bria3.exe -action=importcsv?file=<file path>
- Bria3.exe -action=exportcsv?file=<file path>

- Bria3.exe -action=getstatus?file=<file path>
- Bria3.exe -action=setstatus?status=<status>
- Bria3.exe -action=quit;

From a Hyperlink

In addition you can start calls from URL links via the MIME type "sip:". For example:

<a href = “sip:jsantos@domainA.com>

Where jsantos@domainA.com is a SIP address.