

Overview

Open Communications

libjingle

Developer Guide

Changelist

Important Concepts

How libjingle

Applications Work

Creating a libjingle Application

Signing In to a Server

Sending and Querying Presence

Set Up the Session Management Pathway

Making and Receiving Connections

Scenarios

Sample Applications

File Share Application

Voice Chat Application

Reference

BasicPortAllocator Class

Call Class

Codec Struct

ChannelManager

Call Class

A wrapper object in the voice chat example that bundles groups of **Session** objects into a single chat connection. It provides wrapper methods to handle adding or removing sessions, muting, and monitoring. It also manages creating a new **VoiceChannel** object (which wraps a **P2PTransportChannel** object and a **MediaChannel**) and associating that with each session. This object is created by **PhoneSessionClient**. To destroy it, call **PhoneSessionClient::DestroyCall** (the **Call** object calls this automatically when a **Session** is removed and the **Session** count reaches zero). The methods shown are not thread-safe, and can be called on any thread.

Syntax

```
class Call : public MessageHandler,
            public sigslot::has_slots<>
```

Methods

The following public methods are exposed by **Call**.

Name	Description
void AcceptSession (Session * <i>session</i>)	Accepts an incoming connection request. <i>session</i> is the Session sent out previously by SignalSessionState .
Call (PhoneSessionClient * <i>session_client</i>)	Constructor. Creates a random ID to identify this object.
~Call	Destructor. This method destroys all the Session objects that it holds a pointer to.
uint32 id ()	Returns a random ID assigned to the object (this is not the JID).
Session* InitiateSession (const buzz::Jid & <i>jid</i> , vector<XmlElement*> * <i>extra_xml</i>)	Called by CallClient::MakeCallTo to explicitly make a call to someone. It sends a connection request and a PhoneSessionDescription (a list of codecs) to the other party, and starts generating local candidates. <i>extra_xml</i> is just an extra blob of arbitrary XML to send along with the stanza sent to the other computer (note that this is not used for session description or session info stanzas; this parameter is not used in any of the example code).

Class	void Mute (bool <i>mute</i>)	Mutes the sound in all the Session objects.
Connection Class	bool muted ()	Whether or not the sound is muted in the Session objects.
FileStream Class		
HttpPortAllocator Class	void OnAudioMonitor (VoiceChannel <i>*channel</i> , const AudioInfo & <i>info</i>)	Called by the the Session VoiceChannel objects periodically with audio monitoring data. Sends SignalAudioMonitor . This is not called unless StartAudioMonitor is called.
MediaChannel Class		
MediaEngine Class	void OnMessage (Message <i>*message</i>)	Called with commands from other threads to enable multithreading. The only message handled is to call SessionClient::DestroyCall when the object no longer contains any Session objects.
MemoryStream Class		
P2PTransportCha...		
Port Class	void RedirectSession (Session <i>*session</i> , const buzz::Jid & <i>to</i>)	Sends a redirect reply to a session connection request. This is a Session sent out previously by SignalSessionState .
PresenceOutTask Class	void RejectSession (Session <i>*session</i>)	Rejects a session connection request for the session sent out previously by SignalSessionState .
PresencePushTask Class	std::vector<Session *> & sessions ()	Returns the list of Session objects managed by this call.
Session Class	void StartAudioMonitor (Session <i>*session</i> , int <i>cms</i>)	Causes SignalAudioMonitor signals to be sent. This is not called in libjingle as shipped.
SessionClient Class	void StartConnectionMonitor (Session <i>*session</i> , int <i>cms</i>)	Causes SignalConnectionMonitor signals to be sent. This is not called in libjingle as shipped.
SessionManager Class		
SessionSendTask Class		
StreamEvent Enumeration	void StopAudioMonitor (Session <i>*session</i>)	Ends the sending of SignalAudioMonitor signals.
StreamInterface Class	void StopConnectionMonitor (Session <i>*session</i>)	Ends the sending of SignalConnectionMonitor signals.
StreamResult Enumeration	void Terminate ()	Destroys all the sessions by calling TerminateSession on each Session .
StreamState Enumeration		
Task Class	void TerminateSession (Session <i>*session</i>)	Calls Session::Terminate on each Session , which releases its resources.
VoiceChannel Class		

Signals

XmppClient Class	
XmppEngine Class	SignalAddSession < Call *, Session * > Called when AddSession is called and a session is added. No objects currently subscribe to this signal.
XmppPump	
XmppTask Class	SignalRemoveSession < Call *, Session * > Called when RemoveSession is called and a session is removed. No objects currently subscribe to this signal.
License	
Google Talk XMPP Extensions	SignalSessionState < Call *, Session *, Session::State > Called when a Session sends state information. It is very important to subscribe to this signal because it alerts the application when call connection requests are received, when
User Settings	

- Off the Record Chats
- Jingle Server
- Discovery
- Gmail Notifications
- Shared Status
- Messages
- Extended Contact
- Attributes
- JID Domain Discovery
- Voicemail
- OAuth 2.0
- Authorization

Additional Resources

- Other IM Clients
- Accessories
- Google Talk Blog

calls are terminated, or other important events. Subscribe to this signal as soon as **PhoneSessionClient** sends **SignalCallCreate**.

SignalSessionError< Call *, **Session** *, **Session::Error** >

Called when a **Session** object sends an error message.

SignalConnectionMonitor< Call *, **Session** *, const std::vector< **ConnectionInfo** > & >

Sends periodic connection monitoring information. This signal won't be sent unless you call **StartConnectionMonitor**.

SignalAudioMonitor< Call *, **Session** *, const **AudioInfo** & >

Sends periodic audio monitoring information. This signal won't be sent unless you call **StartAudioMonitor**.

Friends

- PhoneSessionClient

Attributes: public

Declaration file: talk/session/phone/call.h

All rights reserved.

Last updated March 23, 2012.





 **Follow**

1.7M

Explore

Products

Showcase

Events

Communities

Jobs

Connect

Blog

Google+ Community Groups

YouTube Channel

Report an Issue

Programs

Experts

Startups

Women Techmakers

Top Products

Ads

Analytics

Android

Apps

Cast

Chrome

Cloud

Glass

Google+

Maps

Wallet

YouTube