

# SocialSocket の提案

水野 佑樹<sup>†1</sup> 新城 靖<sup>†2</sup>  
佐藤 聡<sup>†3,†2</sup> 中井 央<sup>†3,†4</sup>

## 1. はじめに

近年, SNS (Social Networking Service) を多くのユーザが利用している. IM (Instant Messenger) も SNS の一種であるといえる. SNS が提供するユーザ認証機能や通信機能を利用したアプリケーションが開発されている.

いくつかの SNS にはそのようなアプリケーションのための通信用 API (Application Programming Interface) が提供されている. しかし, それらは一般的に個々の SNS に対して深い知識が求められ, 使用方法も大きく異なるため, 複数の SNS をサポートすることは困難である. また, 特定のプログラミング言語にしか API が提供されていない場合もあり, その場合は他のプログラミング言語で利用することが難しい. 複数のアプリケーション間通信用 API を同じようなインターフェースで利用できるような仕組みがあると便利である.

本研究では *SocialSocket* を提案する. *SocialSocket* は, 複数の SNS の API をラッピングするライブラリである. 複数のプログラミング言語から利用することができる. *SocialSocket* は BSD Socket と類似のインターフェースをもち, ユーザは使い慣れた Socket インターフェースを利用してアプリケーションを記述することができる. また, BSD Socket を用いて書かれたアプリケーションを容易に SNS に対応させることが可能である.

## 2. アプリケーション間通信用 API

本研究では, 次のような SNS で利用可能なアプリケーション間通信に利用可能な仕組みを対象として *SocialSocket* を実装する<sup>☆</sup>.

<sup>†1</sup> 筑波大学システム情報工学研究科

<sup>†2</sup> 筑波大学システム情報系

<sup>†3</sup> 筑波大学学術情報メディアセンター

<sup>†4</sup> 筑波大学図書館情報メディア系

<sup>☆</sup> 多くのユーザが利用する Skype においてもアプリケーション間通信用 API が提供されていたが, 2013 年 12 月に提供が中止

## 2.1 XMPP

XMPP (eXtensible Messaging and Presence Protocol) はインスタントメッセージのためのオープンなプロトコルである. かつては Jabber と呼ばれていた. RFC6120 および RFC6121 で定義されている. メッセージの転送モデルは電子メールと類似しており, ユーザが XMPP サーバを用意して運用することも可能である. 送信されたメッセージは, 複数のサーバを経由して転送される. Facebook, Google talk, iChat, Yahoo!メッセージなど XMPP と互換性のあるサービスが提供されている.

## 2.2 Jingle

Jingle は VoIP (Voice Over IP) やビデオ通話のために P2P (peer-to-peer) 通信の機能を実装した XMPP の拡張である. P2P を利用するために NAT (Network Address Translation) トラバーサル機能が含まれている.

## 2.3 Google+ Hangouts API

Google+ Hangouts は Google 社の提供するビデオチャットサービスである. Google+ Hangouts API は, Google+ Hangouts のネットワークを利用して外部のアプリケーション同士が通信できる機能を持っている.

## 3. SocialSocket の設計

### 3.1 構成

*SocialSocket* を実装するプログラムの構成図を図 1 に示す. *SocialSocket* は, 大きく分けて Network modules, *SocialSocket* interface および Bindings の 3 つの部分から構成されている.

Network modules および *SocialSocket* interface の実装には, C++を用いる. これには利点が 2 つある. 第 1 に, 多くのプログラミング言語において C++へのバインディングを作成することが比較的容易であることである. 多くのプログラミング言語で, それぞれ *SocialSocket* を作成するのは困難であるため, C++

されたため, 本研究ではサポートしないことにした.

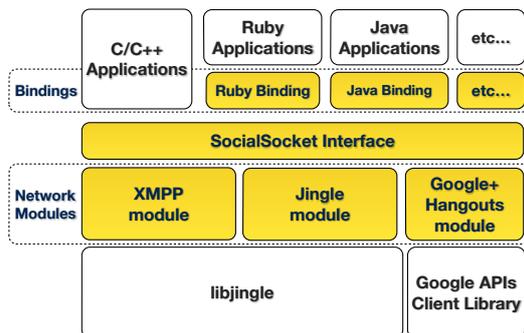


図 1 プログラムの構造

で作成したライブラリを他のプログラミング言語から呼び出す形をとることとする。第 2 の利点として、豊富なライブラリが使用できることが挙げられる。Jingle を利用するために必要な libjingle は C++ によって開発されているため、C++ から利用するのが妥当である。

### 3.2 API

SocialSocket は次のような API を持つ。引数に “\*” がついているものは、結果を受け取るための引数であることを示す。

**socket(domain, type, protocol)** ソケットを生成する。domain には、SocialSocket を表す PF\_SOCIAL\_SOCKET を入れる。type には IP における TCP (Transport Control Protocol) に相当する SOCK\_STREAM または、UDP (User Datagram Protocol) に相当する SOCK\_DGRAM を選択する。protocol には、プロトコルを表す SOCIALPROTO\_XMPP, SOCIALPROTO\_JINGLE または SOCIALPROTO\_HANGOUTS 等を入れる。

**bind(socket, sns\_account)** ソケットをバインドする。このとき、SNS のアカウント情報を引数に取る。これによって、SNS へのログインが行われる。

**getuserlist(sns\_account)** 連絡先リストなどを取得する。

**listen(socket)** 接続待ちにする。

**accept(socket, \*dstUser)** サーバ側において接続を受け付ける。connect してきたユーザの情報を返す。

**connect(socket, dstUser)** クライアント側において指定したユーザに接続する。

**recv(socket, \*buffer)** 受信して、buffer に書き込む。

**send(socket, buffer)** buffer の内容を送信する。

**close(socket)** ソケットを閉じる。

### 3.3 Network module

Network modules は、XMPP, Jingle および Google+ Hangouts の 3 種類に向けたものを開発している。XMPP, Jingle においては、両者に対応したライブラリである libjingle を使用する。このライブラリは Google talk や WebRTC で使用するために作成されたライブラリで、非常に複雑で巨大なライブラリである。Google+ Hangouts においては、Google APIs Client Library を用いて実装する。

本研究により、これらのライブラリの機能を簡単に利用可能なる。

## 4. 関連研究

Liebeherr らによる研究<sup>1)</sup> では、Overlay Network のプログラミングを容易にするため、Overlay Socket を設計している。これは、特定の Overlay Network に依存しない設計になっており、個別の Overlay Network に対して深い知識を必要としない。本研究では SNS において提供されているアプリケーション間通信用 API に対して、個々の SNS に依存しない Socket インターフェースを提供している。

## 5. おわりに

SNS におけるアプリケーション間通信用 API を統一的に使うためのインターフェースである、SocialSocket を提案した。現在、XMPP および Jingle を利用するためのライブラリである libjingle を利用するための調査を行い、ユーザのログイン処理、プレゼンスの通知、データを送信する処理までの実装が完了している。

今後の課題として、SocialSocket の実装がまだ終わっていないため、実装を進めたい。また、セキュリティの観点から通信の暗号化や、アクセス制御についても検討していきたい。SNS のアカウントを使用するため、そのログイン情報の管理についてもセキュリティを保った上で利便性の高い方法を検討したい。

## 参考文献

- 1) Liebeherr, J., Wang, J. and Zhang, G.: Programming Overlay Networks with Overlay Sockets, *Group Communications and Charges. Technology and Business Models*, LNCS, Vol. 2816, Springer Berlin Heidelberg, pp. 242–253 (2003).