

チャットボットの実装・実行の前に必要な準備

チャットボットのコンパイルや実行の前準備として、環境変数 CLASSPATH への Smack API の追加、チャットサーバと動作確認用のチャットクライアントのインストール、および、Yahoo! アプリケーション ID の取得が必要である。

CLASSPATH への Smack API 追加 Smack API はチャット用のプロトコル XMPP に準拠した Java のライブラリであり、

<http://www.igniterealtime.org/projects/smack/>

から入手可能であるが、CD-ROM にも Smack API 3.1.0 を収録した。ソースコードのコンパイルや実行の前に、CD-ROM の Source/Chapter3/NLP をハードディスク上のどこか（下記の例では/home/ai）にコピーし、以下のように環境変数 CLASSPATH を設定する。

```
% cp -r /cdrom/Source/Chapter3/NLP /home/ai/
(シェルが csh, tcsh の場合)
% setenv CLASSPATH ${CLASSPATH}:/home/ai/NLP/smack/smack.jar:/home/ai/NLP/smack/smackx.jar:.
(シェルが bash の場合)
$ export CLASSPATH=${CLASSPATH}:/home/ai/NLP/smack/smack.jar:/home/ai/NLP/smack/smackx.jar:.
```

ただし、ディレクトリ名などは各自の環境にあわせて読み替えること。上記 setenv や export の行を、シェルの設定ファイル（~/.cshrc や ~/.bashrc）に書いておけば、その都度 CLASSPATH を設定し直す手間を省くことができる。また、Unix と Macintosh ではパスの区切り文字が : (コロン) であるのに対し、Windows の場合は ; (セミコロン) であることに注意。このような CLASSPATH の設定をしておいて初めて、以下のようにコンパイルできるようになる。

```
% cd /home/ai/NLP
% javac Chatbot.java -encoding sjis
```

サーバ Openfire とクライアント Spark の用意 本節のチャットボットを動かすためには、XMPP に準拠したチャットサーバとチャットクライアントが必要である。なお、本節で紹介したプログラムは、それぞれ Java で実装されたオープンソースのサーバ Openfire 3.6.4、クライアント Spark 2.5.7 を用いて動作確認した。Openfire と Spark に関する準備の手順は以下の通りである。

1. チャットサーバ Openfire とチャットクライアント Spark をそれぞれ以下の URL から入手し、インストールする。
 - Openfire: <http://www.igniterealtime.org/projects/openfire/>
 - Spark: <http://www.igniterealtime.org/projects/spark/>
2. Openfire を起動する。Openfire のポート番号は、デフォルトの 5222 を変更しないこと。
3. Openfire 上にチャットボットおよびユーザのアカウントを作る。Spark を用いると簡単に作ることができる。
4. Chatbot.java にサーバやアカウントの情報を記述する。具体的には以下のように、チャットサーバ Openfire の URL (19 行目)、チャットボットのログイン名 (23 行目)、チャットボットのログインパスワード (27 行目)、およびユーザのログイン名 (31 行目) を記述する。もしプロキシ設定が必要であれば、プロキシサーバの URL (34 行目) とポート番号 (37 行目) も指定する。

```

11: public class Chatbot extends AbstractChatbot {
12:     :
17:     // チャットサーバの URL
18:     // (XMPP サーバが動いているホスト名に変更する)
19:     public static String XMPP_SERVER = "localhost";
20:
21:     // チャットボットのログイン名
22:     // (好きな名前に変更する)
23:     private static String BOT_NAME = "your_bot_name";
24:
25:     // チャットボットのログインパスワード
26:     // (設定したパスワードに変更する)
27:     private static String BOT_PASS = "apple";
28:
29:     // チャットボットと話すユーザのログイン名
30:     // (好きな名前に変更する)
31:     private static String USER_NAME = "your_name";
32:
33:     // プロキシサーバの URL (必要なら設定する)
34:     public static String PROXY_HOST = "";
35:
36:     // プロキシサーバのポート番号 (必要なら設定する)
37:     public static String PROXY_PORT = "";
38:     :

```

Yahoo! アプリケーション ID の取得 Yahoo! JAPAN のテキスト解析 Web API を用いるための準備の手順を以下に示す。

1. 下記 URL から Yahoo! デベロッパーネットの「アプリケーション ID」を取得する。

http://e.developer.yahoo.co.jp/webservices/register_application

2. 取得したアプリケーション ID を、以下のように Chatbot.java に記述する (15 行目)。

```

11: public class Chatbot extends AbstractChatbot {
12:     :
13:     // Yahoo Web API のアプリケーション ID
14:     // (自分で取得したアプリケーション ID に変更する)
15:     private static String YAHOO_APP_ID = "Y.6xN_K.....";
16:     :

```