



図 1: マルチユーザ・フレームワーク Mug

グループである。

○ **MugUserInfo** アプリケーションはそれ固有のユーザ情報を、ユーザ情報を表す `MugUserInfo` のサブクラスとして定義できる (前述の機能 (2))。 `MugUserInfo` はユーザ情報の属性の名前とその型 (クラス) の表を持っている。 `MugUserInfo` のインスタンスは属性表で定められた型のインスタンスを値としてを保持する。 `MugUserInfo` は、ユーザ ID 番号, ユーザ名, 所属する公開グループ名, ユーザのホスト名を標準の属性として持っている。アプリケーションは、固有の属性を新たに追加したり, 削除できる。

アプリケーションがユーザ情報を利用する場合, 自動的に取得する機構の他に, ユーザ情報の変更をイベントとして受け取る機構も必要となる。 Mug では Java の AWT (*Abstract Window Toolkit*) のイベント処理と同じ委譲型のイベント処理ができる。 `MugUserInfo` の属性の追加・削除, 各インスタンスに対する値の変更のイベントを, アプリケーションの登録したイベントリスナ (*Event Listener*) が処理できる。

○ **MugClient** Mug 上のアプリケーションは `MugClient` (図 2) を共通の GUI として利用できる (前述の機能 (3)(4))。ユーザは, `MugServer` へのログイン・ログアウトをここで行なう。ユーザがログインすると, 右側の「すべてのユーザ」のサブウィンドウに現在ログイン中のユーザの一覧が表示される。



図 2: アプリケーションを組み込んでいない MugClient (ユーザ “ryo” のウィンドウ)

「公開グループ」メニューで, 公開グループへの参加, 脱退ができる。この時, アプリケーションは, イベントリスナに「公開グループに参加しているユーザの顔写真を表示する」などの特別な処理を行わせることも容易である。また, 右側のサブウィンドウの上部にあ

るツールバーを操作することにより, 非公開グループを新たに作成したり, 削除できる。

各サブウィンドウをアクティブにして, 左下のチャット入力欄に文字列を入力することで, サブウィンドウに表示されたユーザとチャットを行なう。サブウィンドウ内の一覧からユーザを選択する (複数も可) ことにより, 指定したユーザだけとのチャットも可能である。

各サブウィンドウのユーザ属性名をマウスでクリックすることにより, 属性の値を鍵としたユーザー一覧の整列ができる。また「定型文」メニューで, 定型文を編集したり, その定型文をチャット入力欄に張り付けられる。

○ **MugServer** `MugServer` はユーザのログイン・ログアウトの管理や, ログイン中のユーザ情報の管理を行なう (前述の機能 (4))。また, ある `MugClient` のユーザ情報の更新やチャットのメッセージなどを他の `MugClient` にマルチキャストする。アプリケーションは, `MugServer` のサブクラスを定義し, `MugServer` のメソッドをオーバーライド (*override*) することにより, アプリケーション固有の処理を行なえる。例えば, ユーザがログインした時に「そのユーザを待っている他のユーザにそれを知らせる」などの特別な処理をしたい場合は, `login` メソッドをオーバーライドすればよい。

5. 実装とアプリケーション

Mug は, JDK (*Java Development Kit*) 1.1, Swing 1.1, Java 分散オブジェクト・プログラミング環境 `RyORB`[2] を用いて実装した。

Mug に組み込むアプリケーションとして, インターネットのような広域ネットワーク上の不特定多数のユーザ同士が遠隔地間で音楽セッションを行なえるシステム `Open RemoteGIG`[3] を開発した。Mug により, 演奏相手を発見する環境や打合せ用のチャット機能を容易に実現でき, ユーザが使用する楽器の種類のような `Open RemoteGIG` 固有の情報を `MugClient` の中でシームレスに表示できた。また, その GUI は, 他のアプリケーションに特化して開発されたマルチユーザ環境と同等の使いやすさが得られた。

6. まとめと今後の課題

本稿では, マルチユーザ環境を必要とするアプリケーションのためのマルチユーザ・フレームワーク Mug について述べた。今後はより多くのアプリケーションを Mug に組み込む予定である。

参考文献

- [1] Toru Ishida, et al.: *Community Computing, Collaboration over Global Information Networks*, JOHN WILEY & SONS (1998).
- [2] 根山 亮: *RyORB* ホームページ, <http://www.info.waseda.ac.jp/muraoka/members/ryo/RyORB/index-j.html>
- [3] 後藤 真孝, 根山 亮: 不特定多数による遅延を考慮した遠隔セッションシステム, 情処研報, Vol.98, No.74, 音楽情報科学 98-MUS-26-14 (1998).