

仮想の“音の部屋”による コミュニケーション・メディア voiscapex における ポリシーベース・セッション制御

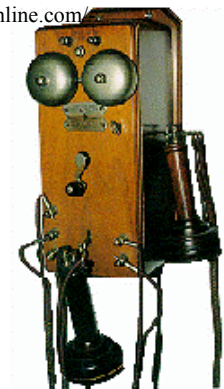
金田 泰
(日立製作所 システム開発研究所)

電話は停滞している

■ 電話の基本インタフェースは発明以来まったく変化していない。

- ◆ 電話は Alexander Graham Bell によって 1876 年 (約 130 年前!) に発明された。
- ◆ 基本ユーザインタフェース「相手に接続し、1 個のマイクと 1 個のスピーカを使用して 1 対 1 で会話し、接続をきる」は変化していない。
- ◆ 電話網の技術的制約のためにインタフェースを改善できなかった。
 - 基本は回線交換網 – かたいネットワーク
 - 回線の接続・切断が必要である。
 - 1 対 1 という制約もはずせなかった。
 - ヒューマン・ファクタによって基本のインタフェースきまっているのではない。
 - インタフェース上の制約の例
 - 切断中は相手の状態がまったくわからない。
 - 3 人以上で会話できない。

1878 年ころの電話器
(<http://www.atcaonline.com/phone/coffin.html>)



停滞は打破される

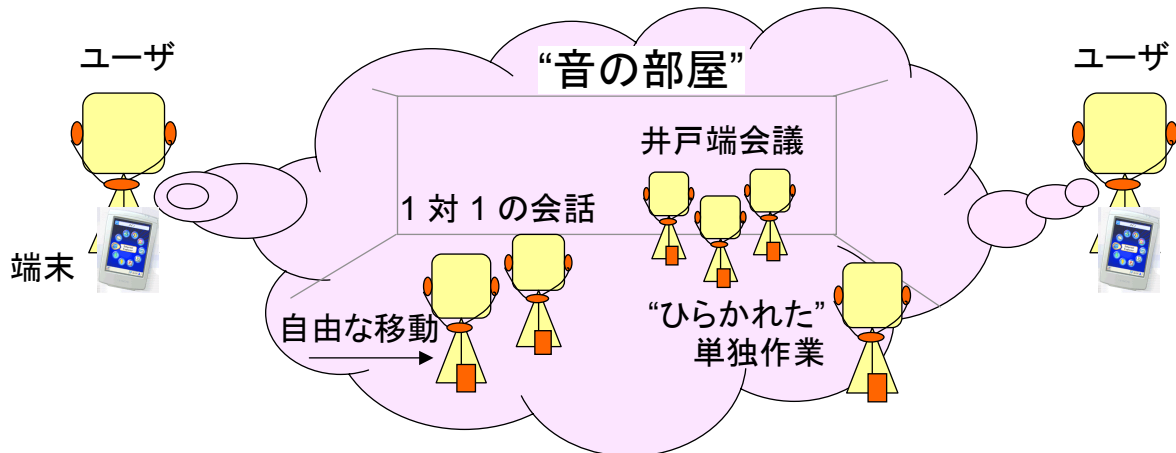
- 現在, インターネットが電話網にとってかわろうとしている.
 - ◆ 現在 IP 電話によって固定電話がおきかえられつつある.
 - ◆ モバイル IP 環境の普及によって携帯電話も IP 化する.
 - ◆ 現在の IP 電話は不必要に制約されたインタフェースを踏襲しているが, この状況は急速に変化するとかんがえられる.
- 将来, 電話は制約のない新メディアによって置換されるだろう.
 - ◆ 常時接続により回線の接続・切断は不要になる.
 - インターネットはパケット交換網なので常時接続が基本.
 - 常時, すこしずつ情報をおくりつづけることができる.
 - ◆ 多者間の自由な会話が可能になる – “会議” メディアになる.
 - 1 対 1 という制約も存在しない.
 - ◆ 人間の聴覚能力をもっといかすことが可能になる.
 - 帯域幅の制約もゆるむので, 1 チャンネルに限定する必要もない.

電話にかわるインタフェース?

- 電話にかわるべきメディアを voiscap とよぶ
 - ◆ Voiscap の語源
 - 作曲家 Murray Schafer による用語 soundscape にならう.
 - Soundscape = 音によってつくられる風景
 - Voiscap = 声の風景, 声景

想定する voiscap のすがた

■ 音によって 3D 仮想空間をつくり、そのなかで会話する。



■ リアルタイム通信と 3D オーディオのくみあわせ

- ◆ 3D オーディオを使用する理由: 3人以上で話をするときは、方向感・距離感が重要。
 - それによって話者をくべつしやすい。
 - 2人以上が同時に話をして、カクテル・パーティ効果によって分離できる - 人間の聴覚能力をいかす。

Voiscap における会話のながれ [案]

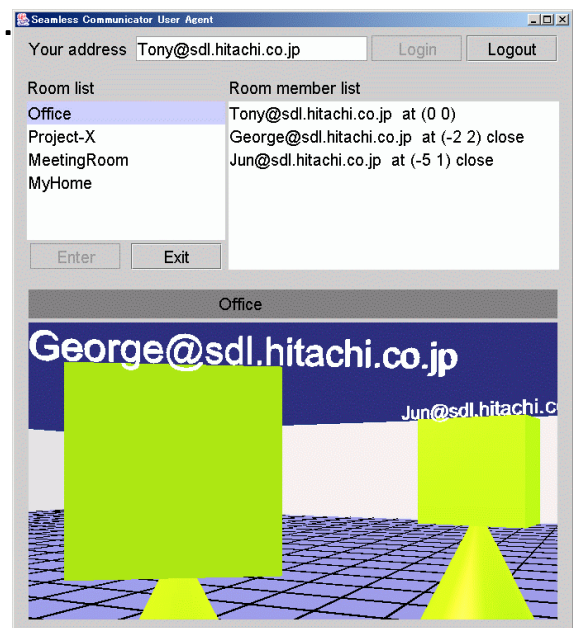
■ ログインして部屋リストを表示する。

■ 部屋を選択する。

- ◆ 右図では 4 つの部屋がある。
 - オフィス (Office)
 - プロジェクト X (Project-X)
 - 会議室 (MeetingRoom)
 - 家 (MyHome)

■ 入室すると部屋内が表示される。

- ◆ 前方の様子がグラフィクス表示される。
- ◆ 部屋にいるメンバーのリストが表示される。



■ 部屋内で移動・回転して会話相手に接近する。

- ◆ 移動・回転にはポインティングデバイス (現在はマウス) を使用。

ポリシーによる通信・表示の制御

■ Voiscap ではポリシーによる通信や表示の制御が有効 (?!)

- ◆ ポリシーとは、特定の条件がなりたつときにどのような通信や表示の動作をするかを記述した規則またはそのあつまりのこと。
 - 通常は if-then 型の規則を使用する。
- ◆ ポリシーは端末に内蔵する。

■ Voiscap におけるポリシー制御の目的

- ◆ 通信量削減
 - 部屋内の全ユーザと常時通信するのが原則 – 通信量が膨大になりうる。
 - 低能力の端末では通信量や相手の制限が必要 – 端末ミキシング (3D 化) するとき。
 - 回線容量などによっては通信量の制限が必要
- ◆ プライバシー保護
 - 部屋内で一定値以上の距離にいるユーザとの接続を切断 / 通信内容を制限するのがよい。
 - 例: 相手との距離によって音声の明瞭度や帯域幅を制御する。

ポリシーによる通信・表示の制御 (つづき)

■ ポリシーの例

- ◆ 「部屋内でユーザどうしの距離が一定値以下になったときに接続し一定値をこえたときに切断する」
- ◆ このポリシーによって、相手の存在に気がつかないほど遠方のユーザによって話をぬすみきかれることがさけられる。
 - プライバシー保護, 通信量削減に有効。
- ◆ 現在プロトタイプに実装しているポリシーはこれだけ。

■ ポリシーの調停

- ◆ 通信相手がことなるポリシーをもっていると調停が必要。
- ◆ 現在の方式では、各端末が P2P で通信相手ごとに個別に調停する。
- ◆ 相手がことなるポリシーをもつときは、よりつよいプライバシー保護を実現するポリシーが双方に適用される。
 - 例: ユーザ間距離による接続・切断に関するポリシーを双方がもっているときは、みじかいほうの距離にあわせられる。

距離にもとづくポリシーにおける調停の例

■ ポリシーの例

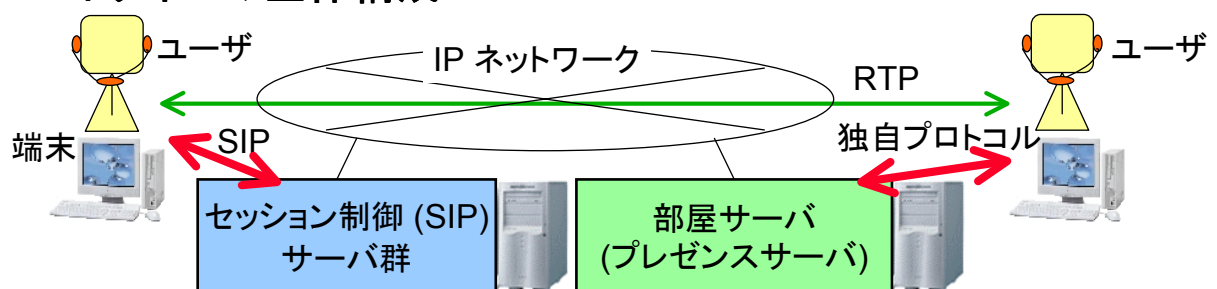
- ◆ if 距離 $< d$ then 接続
- ◆ if 距離 $> d + \alpha$ then 切断

■ 調停法

- ◆ 状況: ユーザ u_1, u_2 間の接続において接続距離 d_1, d_2 ($d_2 > d_1$) がことなるとき
- ◆ 接続時刻の決定
 - みじかいほうの距離 (d_1) によってきまる.
 - 通信シーケンス
 - 距離が d_2 以下になると u_2 が u_1 に INVITE メッセージをおくる.
 - このとき u_1 は接続距離に達していないので、これを拒否する.
 - 距離が d_1 以下になると u_1 が u_2 に INVITE メッセージをおくる.
 - このとき u_2 にとっても接続距離に達しているため、通信が成立する.
- ◆ 切断時刻の決定
 - 接続と同様に、みじかいほうの距離 ($d_1 + \alpha$) によってきまる.

Voiscap プロトタイプ開発

■ プロトタイプの全体構成



- ◆ デスクトップ PC 上で実験した.

SIP = Session Initiation Protocol
RTP = Real-time Transport Protocol

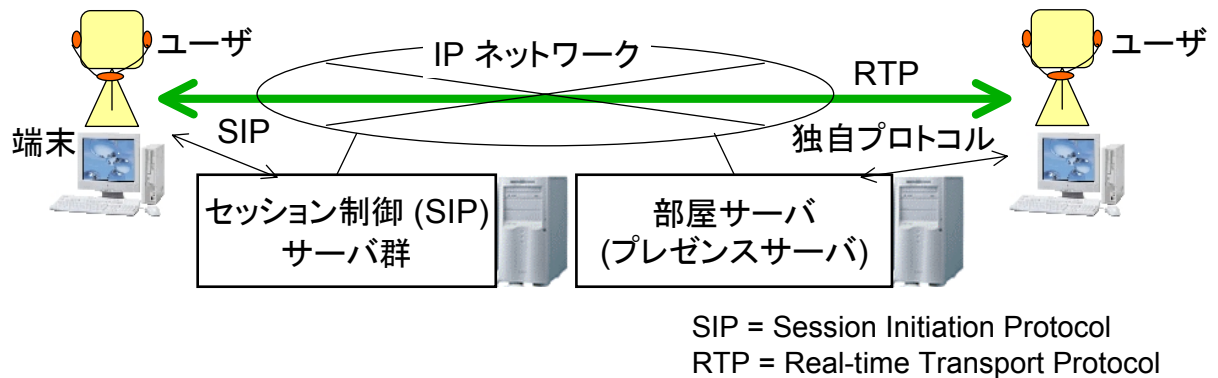
■ サーバ群の機能・構成

- ◆ セッション制御 (SIP) サーバ群
 - 距離にもとづくセッションの制御 (自動接続・切断) などをおこなう.
 - SIP プロキシ, 登録サーバ (registrar), 場所サーバ (location server) をふくむ.
- ◆ 部屋サーバ (プレゼンスサーバ)
 - 部屋の属性とその利用者 (ネットワーク接続, 在・不在, 位置など) を管理.
 - このプロトタイプにおいては SIP / SIMPLE でなく独自プロトコルを使用.

Voiscape プロトタイプ開発 (つづき)

■ プロトコルとコーデック

- ◆ セッション制御 (サーバベース) には SIP 使用.
- ◆ 音声はユニキャスト P2P で RTP により通信.
- ◆ 音声は 8000 Hz でサンプリング (ITU-T G.711 u-law 64 kbps)



部屋サーバの機能

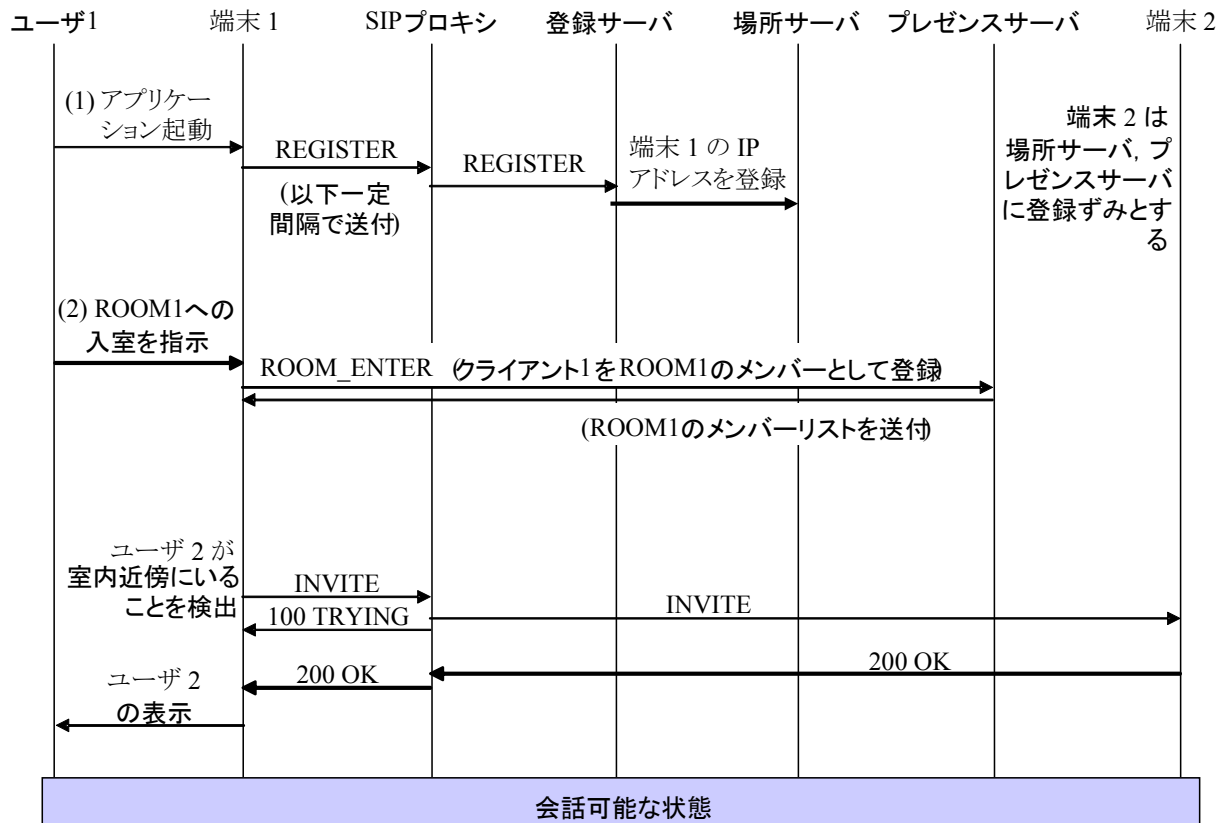
■ ユーザの位置情報の把握にはポーリングを使用

- ◆ 端末が時間間隔をきめてサーバに報告する.
- ◆ イベントにもとづく方法もかんがえられる.
 - SIP subscribe/notify (SIMPLE) の使用など:
使用した SIP スタックには実装されていなかった.

■ 部屋サーバのプロトコル

- ◆ 要求-応答 型のプロトコル
- ◆ 要求メッセージは 4 種類
 - ROOM_ENTER (部屋の生成 and/or ユーザの入室)
 - ROOM_REMOVE (ユーザの退室)
 - PRESENCE_REFRESH (双方向のプレゼンス更新. 部屋リストもふくむ)
 - ROOM_DESTROY (部屋の抹消)

プロトタイプにおける接続までのシーケンス

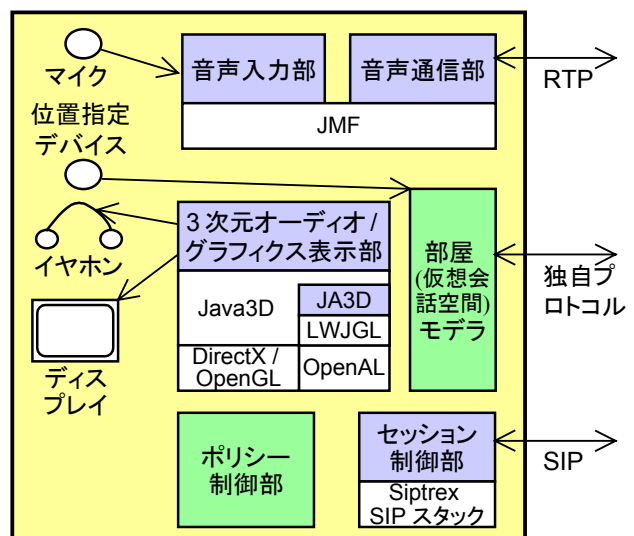


ユーザ・エージェント UA の構造

■ UA は部屋サーバからの情報により音声を 3D 化.

■ 使用したソフト – Java API

- ◆ SIP スタック: UCL の Siptrex
 - NIST SIP 旧版をベースに開発
- ◆ 音声入出力・通信: JMF (Java Media Framework)
- ◆ 3D グラフィクス: Java 3D
- ◆ 3D 音声: Open AL など
 - JMF には 3D 音声機能もあるが, Java 3D とくみあわせて使用できなかった.



■ HRTF (Head Related Transfer Function) 機能を使用.

- ◆ 左右だけでなく前後や上下方向もあわせた方向感を表現する.
- ◆ HRTF とは人間の頭部周辺の音響特性のこと.

セッション制御における問題: glare

- INVITE メッセージを受信する相手が, 受信前に INVITE メッセージを送信していることがある.
 - ◆ 電話でいえば双方が同時に受話器をあげたケース:
 - 通常は双方が話し中になって通信は確立されない.
 - ◆ Voiscap ではこのケースでも通信が確立されなければならない.
- 現在の解決策: どちらの INVITE メッセージに対しても 200 OK メッセージを返信し, さらに ACK メッセージをおくる.
 - ◆ SIP メッセージングに関しては 2 重の接続関係が生じる.
 - ◆ RTP の接続は 1 重 (ただし双方向) にする.
 - ◆ この方法では 2 個の UA に関してメッセージが対称に交換される.
 - SIP サーバは通常のシーケンスを実行するだけである.
 - ◆ BYE メッセージも 2 重化する必要がある.
 - このメッセージングに関与する SIP プロキシがステートフルであるばあいに, そのプロキシがもつ状態をクリアする必要があるから.

セッション制御における問題: glare (つづき)

- 代案: なんらかの方法で 2 個の INVITE メッセージに順序ないし優劣をつけて対称性をやぶり, 一方をとりけす.
 - ◆ SIP においてはユニークな call id が使用されるので, INVITE メッセージの call id の大小によって選択すればよい.

試作の結果と検討

■ 接続にかかる時間の評価

- ◆ 2 台の UA が接続するべき距離に達してから実際に相手の音声がかきとれるまでの時間は約 7 秒.
- ◆ 実用上ながすぎるが, ちぢめる方法はまだわかっていない.
 - 原因は JMF にある (?!)
- ◆ 切断時は交渉をしないため即座にきれる.

■ プレゼンスサーバの実装に関する結果と検討

- ◆ 部屋サーバの負荷がたかい.
 - Intel Pentium III 800 MHz の負荷が 50% をこえることがある.

■ 音質劣化と遅延

- ◆ 遅延は最大で約 6 秒 – 軽減されたがまだ完全に解決していない.

■ リアルタイム処理上の問題点

- ◆ Java 3D を中心に約 70 個のスレッドが生成される – 過大.
- ◆ 音質劣化と遅延にはスレッドのスケジューリング関与の可能性が大.

結言

■ 結論

- ◆ 新メディア voiscap を提案した.
 - リアルタイム通信と 3 次元オーディオ / グラフィクスとをくみあわせた.
 - 人間のコミュニケーション能力と IP ネットワークの常時接続性をいかす.
- ◆ 端末上のポリシーによってセッションを制御することにより, プライバシー保護や通信量削減が可能になる.
- ◆ 接続開始時に 2 重に接続したり “話し中” になったりせずに接続する方法をくふうした.
- ◆ Java によるプロトタイプ開発によりさまざまな課題が抽出された.

■ 今後の課題

- ◆ サーバ / ネットワークの負荷を軽減し常時接続使用を可能にする.
- ◆ 小型化・ウェアラブル化をはかる.
- ◆ 認知的な評価をおこなう.