

修 士 論 文

インフォーマルコミュニケーション 支援プレゼンス機構の設計と実装

指導教官

青山 友紀 教授
森川 博之 助教授



東京大学大学院新領域創成科学研究科
基盤情報学専攻

氏 名

47-36306 小野 孝之

提 出 日

平成 17 年 1 月 31 日

目次

第1章	序論	1
第2章	プレゼンス技術	5
2.1	はじめに	6
2.2	プレゼンスの実現	7
2.3	現状におけるプレゼンスの活用法	9
2.3.1	CSCW	9
2.3.2	インスタントメッセージング	10
2.3.3	関連研究	11
2.4	おわりに	13
第3章	インフォーマルコミュニケーションとプレゼンス	14
3.1	はじめに	15
3.2	コミュニケーションの内容と回数	15
3.3	個人のプレゼンスと環境のプレゼンス	16
3.4	プレゼンス情報の流通	16
3.5	ユーザとシステムの心理的距離	18
3.6	おわりに	20
第4章	ACDC システム	21
4.1	はじめに	22
4.2	ACDC システム構成の概要	22
4.3	情報の取得	25
4.4	情報の伝達と合成	25
4.5	ユーザエージェント	25
4.5.1	通信部 (Network Transaction)	26
4.5.2	保存部 (DB)	28
4.5.3	処理部 (Computation)	28
4.6	おわりに	30
第5章	コミュニケーションに与える影響の評価	31
5.1	はじめに	32

5.2	ACDC システムの実装	32
5.2.1	実装方針	32
5.2.2	フレームワーク	32
5.2.3	プレゼンス情報	40
5.2.4	実空間との入出力連携	45
5.3	実験評価	46
5.3.1	実験の目的と構成	46
5.3.2	結果と考察	47
5.3.3	得られたその他の知見	50
5.4	おわりに	52
第 6 章	結論	53
6.1	本論文の主たる成果	54
6.2	今後の課題	55
	謝辞	56
	参考文献	57
	発表文献	60

目次

2.1	プレゼンスの利用出来ない環境	8
4.1	ネットワーク空間への投影	23
4.2	システム概要	24
4.3	受信者側での情報合成	26
4.4	ユーザエージェントの構成	27
4.5	過去のプレゼンスの DB への記録	29
5.1	スクリーンショット	34
5.2	プレゼンス伝達のパケットの流れ	35
5.3	SIP によるチャットセッションの確立	37
5.4	コミュニケーションプレゼンスの表示例	42
5.5	部屋の雰囲気への伝達	43
5.6	自分の部屋の選択画面	44
5.7	AIBO による情報呈示例	45
5.8	C の各ユーザあたりの平均値	48
5.9	t の各ユーザあたりの平均値	48
5.10	N_m の 1 回あたりの平均値	49
5.11	L_m の 1 回あたりの平均値	49
5.12	N_o の 1 回あたりの平均値	49
5.13	L_o の 1 回あたりの平均値	49

表目次

5.1 DLL の提供する API 一覧	38
5.2 コールバック関数のイベントと引数	40
5.3 測定項目	46
5.4 実験条件	47

第1章 序論



ARPA による学術・軍事目的のネットワークとしてスタートしたインターネットは、時代の経過と共に徐々に接続ホスト数が増大し、ついに 1990 年代に入って商用利用が開始されるに至った。その後はホスト数の増大や帯域幅の拡大が加速的に進行し、今日では我々の生活に欠かすことのできないインフラストラクチャとなっているということは衆目の一致するところである。

一般的なエンドユーザの立場で考えると、ほんの 10 年前は利用出来る帯域は数十 kbps 程度であり、かつ従量課金であるため必要なときだけインターネットに接続する、というのが標準的なスタイルであった。より高速な帯域や、24 時間いつでも接続出来る定額制のネットワークの利用は高嶺の花であり、そのようなネットワークは研究機関や大企業など、ごく一部のユーザしか享受することが出来なかった。

しかし今日、DSL や光ファイバに代表されるネットワーク技術の進歩により、広帯域かつ常時接続されたネットワーク環境が安価に利用出来るようになり、これらはもはや企業や大学だけのものではなく一般家庭においても珍しいものではなくなってきている。このような常時接続されたネットワーク環境を利用し、家にいながらオフィスと同じように仕事を行ういわゆる在宅勤務や、従来は同一のオフィス内でしかなし得なかった作業を物理的に離れたオフィスと協調して行うことへの要求が、今後ますます盛んになると考えられる。

このような、直接相手の顔が見えない環境での協調作業をスムーズに行うために、従来から様々な方法が研究・提案されてきている [6, 21, 22, 24, 26, 35]。それらの多くは「いかにして多くの情報を伝えることでスムーズなコミュニケーションを行うか」「いかにしてネットワークを用いることで物理的な距離を短縮するか」ということに主眼が置かれている。そのような研究も重要であるが、情報が多く伝わりすぎることのデメリットや、ネットワークにより距離が短縮されすぎることのデメリットについても考慮する必要があると、筆者は考える。

地理的に離れた場所にいる同僚と何らかのコミュニケーションシステムを用いて打ち合わせを行う際には、音声だけよりは映像もついていた方が、また映像の解像度やフレームレートはより高い方が質の高いコミュニケーションを行うことができるというのは自明である。このような「必要なときだけつなく」システムにおいては、伝える情報量を増やすことはその目的と良く合致する。しかし、「必要なときだけつなく」という利用形態は、離れた場所にいる相手と会議や打ち合わせなどフォーマルなコミュニケーションを行うのには適しているが、そのようなシステムをそのまま日常のコミュニケーションのうち多くを占める、雑談や一服しながらの立ち話の様なインフォーマルなコミュニケーションに用いることには問題が多い。コミュニケーションのあり方を考えた場合、フォーマルなものと同様にインフォーマルなものもまた重要である。本稿ではこれらのうちインフォーマルコミュニケーションに着目し、これを支援することを目的とする。

インフォーマルなコミュニケーションの特徴としては、以下のような点が挙げられる。まず、フォーマルなコミュニケーションと比較すると、インフォーマルなコミュニケーションは偶発的であり、相手も話題も実際にコミュニケーションが始まるまで不定であるという点が挙げられる [36]。また、日常のインフォーマルなコミュニケーションを重ねることで人間関係はより親密なものとなりうるし、

そのようなコミュニケーションの中から有益なアイデアなどが生まれることも考えられる。さらに、チャットなどのコミュニケーションをとるに至らないまでも、離れた場所にいる仲間がどんなことをしているのか、あるいは離れた場所の部屋はどんな雰囲気なのかとすることが“なんとなくわかる”ことで日常生活を円滑に行うことが期待できる。ほかにも、大学の研究室や会社のオフィスにおいては、そこに所属するメンバーは固定ではなく、年に何回か新人が加わるというのが一般的である。そのような新人と既存のメンバーが協調作業を行うためには信頼関係の醸成が不可欠であり、そのためには普段のインフォーマルなコミュニケーションを積み重ねると共に、日常生活の様子といったものを伝えることで「繋がっている感」を持たせることが有意義であると考えられる。またこのような「繋がっている感」を重視する傾向は、インターネットの発展と共に成長してきた若い世代で顕著であることが報告されており [10, 32]、このような傾向は今後も続くものと考えられる。

上記のような特徴を持つインフォーマルコミュニケーションの支援においては、「必要なときだけつなぐ」のではなく、「常につないでおく」ことが考えられ、常につないでおくことでユーザ自身やユーザの周囲の状況を、プレゼンス情報として他のユーザに広告することが考えられる。このような観点から、本研究では適切な内容のプレゼンス情報を適切な形でユーザに呈示することで、前述の「つながっている感」をユーザが感じ取ることができると共に、適切な時機に適切な話題でコミュニケーションを行うことが期待できるプレゼンスの活用注目する。

プレゼンス情報として常時他のユーザに広告することを考えた場合、情報量の多い通信手段が優れた通信手段であるとは必ずしも言うことができない。確かに、情報を受け取る側にとっては情報量が増えることは歓迎すべき事である。音声だけよりも映像が付加されていた方が、発信者側がどのような状況下に置かれていてどのようなことをしているのかとすることが詳しくわかる。しかし、一般的な双方向のコミュニケーションにおいては受信者は同時に発信者でもある。そして発信者の立場に立って考えると、映像のような情報量の多い媒体で常に自分の状況を他のユーザに広告されるのは、プライバシーや心理的な観点から非常に抵抗が大きい。

このように、常につないでおくことでいわばコミュニケーションを行うことへのハードルを下げることを目的とする場合には、要求される情報量や情報の粒度が、明示的なコミュニケーションを目的として必要に応じてつながれる場合と異なってくるといえる。そこで、本研究では適切に調節された情報量のプレゼンスの伝達を通して、ユーザのインフォーマルなコミュニケーションの支援を行う。プレゼンスの種類は極めて多岐にわたり、それらを全て利用することは現実問題として不可能であり、どの種のプレゼンスを利用するかという点が問題になる。究極的にはある 1 種のプレゼンスを伝えることで本稿で目標としているようなコミュニケーションの支援が出来ることが理想であり、筆者はそのプレゼンスがどのようなものであるかについて検討を行っているが、現時点ではその解を見いだすに至っていない。さらなる検討を有意義なものとするためには、どのような情報をプレゼンスとしてどのような形で他のユーザへと伝達することがインフォーマルなコミュニケーションの支援に効果的であるかを、実際に試用していく中で見極める必要がある。そこで、本研究では様々な種類のプレゼンス情報を扱うことのできるシステムの形態について検討を加えると共に、いくつ

かのプレゼンス情報を利用出来るテストベッドとして実装を行ったアプリケーションについて述べ、そのテストベッドを用いて行った、プレゼンス情報のコミュニケーションに与える影響に関する評価実験の結果について報告する。

本稿の構成は以下の通りである。

第 1 章 序論

第 2 章 プレゼンス技術

第 3 章 インフォーマルコミュニケーションとプレゼンス

第 4 章 ACDC システム

第 5 章 コミュニケーションに与える影響の評価

第 6 章 結論

まず、第 2 章でプレゼンスの実現方法について述べると共に、今日行われているプレゼンスの活用法についていくつかの例を紹介しながら述べる。

第 3 章ではインフォーマルコミュニケーションとプレゼンスの関係について述べ、インフォーマルなコミュニケーションの支援を行うシステムについて検討する際に考慮しなくてはならない項目について述べる。

第 4 章では第 3 章での検討事項をふまえた上で、実際にシステムを構築する場合の手法について述べる。ユーザをネットワーク空間に投影するユーザエージェントを用い、ユーザエージェントを通してのユーザ同士のプレゼンスのやりとりについて述べる。

第 5 章では、プレゼンス情報がコミュニケーションに実際にどのような影響を与えているのかを評価するために行った実験について述べる。まず、実験を行うために実装を行ったシステムについて報告し、次に実験を行った結果について述べる。

最後に第 6 章で本研究の成果をまとめ、今後の展望について述べる。

第2章 プレゼンス技術

2.1 はじめに

インターネットはその誕生以来、様々な形でユーザ間のコミュニケーションに利用されてきた。古くから利用されているコミュニケーションメディアとしてはE-mail やウェブ (World Wide Web: WWW) といったものが挙げられるが、これら旧来のネットワークコミュニケーションは、情報の発信者と受信者の間の同期性という観点から見ると非同期的なものが中心であった。例としてE-mail によるコミュニケーションを見てみると、メールの送り手はメールを送信するときに相手が現在何を行っているかと言うことを考慮することは稀である。ユーザがそれぞれ自分の都合に合わせてメールを受け取り、必要に応じてそれに返信するというのがE-mail の典型的な利用スタイルであり、またこのように相手の状況と関係なく情報を伝えることができるということがE-mail の様な非同期コミュニケーションシステムの利点でもある。

しかし、ネットワークへの常時接続が一般的になるに従って、ネットワークコミュニケーションもその姿を大きく変えつつある。ICQ [14] に代表されるインスタントメッセージング (IM: Instant Messaging) ツールや、携帯電話のショートメッセージサービス (SMS: Short Message Service) においては、送信者が送信したメッセージは通常ほぼ即座に受信者のもとへと届けられる。さらに、当初はテキストのみであったコミュニケーション手段も、昨今ではSkype [27] に代表されるVoIP (Voice over IP) アプリケーションやMSN メッセンジャー [19] のビデオチャットのように音声や動画像など様々なものを利用出来るようになりつつある。

このように、ネットワークを用いたコミュニケーション手段が同期的に行われるようになり、また利用出来るコミュニケーション手段も多様化するに従い、コミュニケーションを始めるにあたって相手がどのような状況に置かれているかと言うことを、あらかじめ何らかの形で知っておきたいという要求が増大するのはごく自然なことと考えられる。相手の状況を知ることが出来れば、利用するコミュニケーションツールをより適切と思われるものに変更したり、あるいはコミュニケーションを行うタイミングを変更するということが実現可能になり、よりユーザにとって利便性の高いシステムが実現出来る。

本章では、まずプレゼンスの基本的な概念と、プレゼンスに求められる基本的な概念について言及する。次に、現状におけるプレゼンスの活用法について述べ、本研究との差異を明らかにする。

2.2 プレゼンスの実現

一般的に述べて、プレゼンスとは“ユーザの状況を伝える情報”と定義することができる。ユーザが置かれている環境についての情報を伝達することで、ネットワークコミュニケーションを円滑に行おうというのがその目的である。そのようなユーザを取り巻く環境についての情報というものは、通常のフェイストゥーフェイスのコミュニケーションにおいては各自が意識的に、あるいは無意識のうちに、様々な五感によって把握しているものである。具体的には、以下のようなタイプに分類することができる。

- 相手が今何をやっているか、どのような場所にいるかと言うようなそれぞれの人間に関する客観的な事実としてとらえられる事柄。
- 暇そうかあるいは忙しそうか、何を考えているのか、機嫌はいいのか悪いのかといったような人間の内面に関するもの。これらの情報は、一般には他人は五感により観察された事柄をもとに主観に基づいた推測を行う。
- 周囲の明るさや温度など、各々の人間の周囲の状況を表し、場合によっては複数の人間で共通の値となるもの。

フェイストゥーフェイスのコミュニケーションにおいては各々の人間がごく自然に取得し、理解しているこれらの情報により、コミュニケーションの開始のタイミングが調節されたり、場合によっては話題の提起も行われる。具体的な例としては、「相手が忙しそうな場合には急ぎでない用件であったら後で話すことにする」ということや、「機嫌が悪そうだからくたらない軽口を叩くのはやめておこう」といったことは我々がしばしば経験することである。

しかし、ネットワークコミュニケーションにおいては、これらの情報はコミュニケーションを開始する前には得ることが出来ない。勿論、コミュニケーションを開始した後ではコミュニケーションに用いるメディアに応じて様々な情報を得ることが出来るが、そもそものコミュニケーションを行うかどうかという状況判断にはそれらの情報は用いることが出来ない(図2.2)。その結果として、「忙しいときにどうでもいい話題で話しかけられる」「席を離れている間にビデオチャットに誘われる」というような、話し手(caller)と受け手(callee)との間での状況認識の不一致により双方に不利益をもたらすような事態が生じる可能性が考えられる。このような事態は、ネットワークコミュニケーションの進歩により、コミュニケーションメディアとして同期性を持つものが大きく台頭してきたことによりもたらされ、旧来のE-mailによるコミュニケーションの時代にはあまり考慮されることの無かった新しい問題である。

このような caller・callee 双方への不利益を解決するための手段として考慮されているのが、いわゆるプレゼンス情報である。フェイストゥーフェイスのコミュニケーションの場合にはコミュニケーションを始める前に様々な形で取得されうる情報を、ネットワークを通じて「プレゼンス情報」と

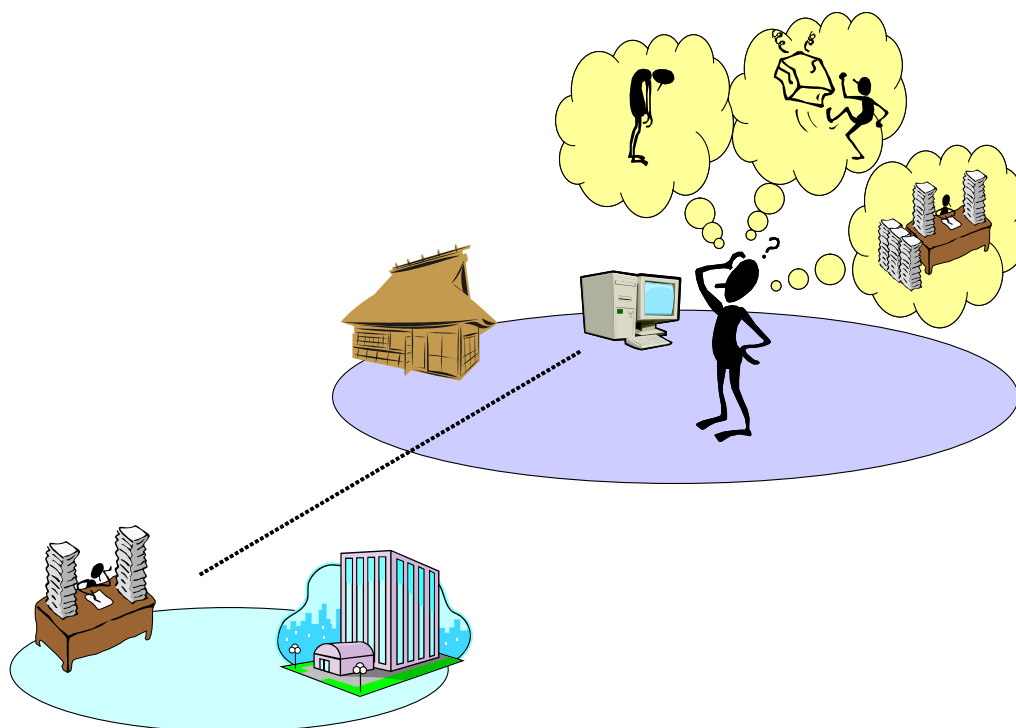


図 2.1: プレゼンスの利用出来ない環境

してあらかじめコミュニケーションが開始される前にネットワーク上の周囲のユーザに広告しておくことで、ネットワークコミュニケーションにおいてもフェイストゥフェイスのコミュニケーションと同様に、ユーザがコミュニケーションを行う機会や話題についての判断材料として用いようとするものである。ユーザの周囲に偏在するあらゆる情報をネットワークを用いて伝送し、離れた場所にいるユーザに適切な形で呈示することにより、離れた場所にいるユーザがどのようなことを行っているのかということはもとより、ユーザがそこに存在しているということ自体を「存在感」として「何となく」伝えられることが期待出来る。

しかし、一言で「ユーザに関する情報」と言ってもその種類は極めて多岐にわたる。また、ネットワークによって伝達を行うことが前提であるので扱う情報は計算機が扱える電子情報である必要があるが、我々の身の回りに存在する様々な情報のうちそのまま計算機で扱える形で存在しているのはごく一部であり、ほとんどの情報はセンサなど何らかの方法によって電子化することではじめて計算機で扱い、ネットワークを用いて伝送することが出来るようになる。また、情報の種類によってはセンサなどを用いても直接得ることが出来ず、得られたデータと過去のデータから推定によってしか求められない情報というものも考えられる。また、そのように多岐にわたる膨大な情報をどのような形でユーザに呈示するのかということも大きな問題であると言える。

このように、プレゼンス情報のネットワークを介したやりとりが実現されることによって、ネットワークコミュニケーションはより豊かなものになり、フェイストゥーフェイスのコミュニケーションと同等なものになることが究極的には期待される。しかし、そのためには解決しなくてはならない問題が数多く存在し、それらを解決することではじめて前述のようなビジョンが実現されうると言える。

2.3 現状におけるプレゼンスの活用法

本節では、現状における様々なプレゼンスの活用法について紹介し、あわせて問題点を明らかにする。

2.3.1 CSCW

プレゼンス情報については、現在様々な場面での利用が検討されているが、そのなかでも Computer-Supported Cooperative Work (CSCW) と呼ばれる分野において特に注目を集めている。CSCW とはその名の通り、複数の人間による協調作業を計算機を用いることで支援しようという取り組みである。

CSCW に分類される取り組みとしては数多くのものであるが、それらは大きく二つに分類することができる。すなわち、協調作業を行うユーザ同士が物理的にすぐそばにいる場合と遠く離れた場所にいる場合であり、これらのうち、プレゼンスが大きく関係してくるのはユーザ同士がフェイストゥーフェイスのコミュニケーションを行うことが不可能な後者である。

協調作業というのは、作業形態がどのようなものであれ作業員同士がコミュニケーションを行うことが必要不可欠である。そこで、ユーザ同士のコミュニケーションをより豊かにするための方法としてコミュニケーションを行う際により多くの情報を伝えることで目的を果たそうという試みは割合古くから行われてきた [3, 8, 13]。これらの試みでは、コミュニケーションのメディアとしてテキストだけではなく音声や映像をあわせて用いるとともに、ユーザを映した画像を周囲のユーザに対して常に公開しておくことで離れた場所にいるユーザ同士を結びつけることも行っている。しかし、これらの試みが行われてから久しいが、今日このような方法が広く一般に広まっているとは言い難い。

このような方法の普及を妨げている要因として、Ellen Isaacs らは以下のような点を挙げている [16]。

- 例え相手ユーザの姿が映像で映し出されても、相手が何に集中しているのかそれだけではわからないという問題
- 各ユーザ毎にカメラを設置することなどコスト面の問題
- 常に自分の姿を他人に見られてしまうというプライバシーの問題
- 純粋に技術上の実装の難しさの問題

これらの観点から、CSCW の分野においては実際にコミュニケーションを行うメディアとして、音声や映像ではなくインスタントメッセージの利点が見直されつつある [20]。次節ではインスタントメッセージを用いた代表的なアプリケーションとして、インスタントメッセンジャーについて説明する。

2.3.2 インスタントメッセージング

一般的に ICQ [14] に代表されるインスタントメッセンジャーでは、あらかじめ直接会ったりメールでやりとりしたりしてユーザ間で交換した ID を使用して、ユーザ同士がテキストベースのチャットによるコミュニケーションを行う。なお、最近では MSN Messenger [19] 等の様に音声通信やビデオチャットなどをサポートしてよりリッチなコミュニケーションを行うことを目指すものもある。これらのインスタントメッセンジャーでは、ユーザが各自の端末上でクライアントソフトウェアを起動してサーバに接続することによってユーザ間の同期的なコミュニケーションが可能になる。各ユーザのクライアントソフトウェア上には、自分があらかじめ ID を登録したユーザのリスト (BuddyList) について、それぞれオンラインでコミュニケーション可能かあるいはオフラインでコミュニケーションがとれない状況にあるかが一覧で示される。自分がオンラインの間に他のユーザがオンラインになった場合には、その情報は瞬時にクライアントソフトウェアに反映される。

このようなインスタントメッセンジャーにおいては、サーバに接続していない間は他のユーザからのメッセージを受け取ることが出来ないため、各ユーザが端末を使用している間はクライアントソフトウェアを常駐させておくというのが一般的な使用方法となっている。しかし、複数のユーザがひとつの端末を共有し交代で使用するという利用スタイルは過去のものになりつつあり、昨今では各ユーザがそれぞれ自分専用の端末を有するというスタイルが普遍的になりつつある。そのような場合では、ユーザは端末の電源を入れたまま他の作業を行ったりあるいは席を離れたりすることが一般的に行われる。このように、端末の前で様々に変化するユーザの状況を離れた場所にいる BuddyList 内の各ユーザに知らせるために、多くのインスタントメッセンジャーでは「退席中」「電話中」「取り込み中」といったユーザの状況を知らせるプレゼンス機能を備えている。また、本来は自分の名前を“スクリーンネーム”としてフリーテキストで記述する機能を用いて、名前と共に自分の置かれている状況や気分などをで書き込むことも一部のユーザの間で行われている。このような傾向は十代の若いユーザの中に多く見られ、また日常的にインスタントメッセージサービスを多く使うヘビユーザほどその傾向が顕著であることが報告されている [10]。

このほか、インスタントメッセージのみが持ち他のコミュニケーションメディアには見られない特性として、インスタントメッセージは同期的なコミュニケーションと非同期コミュニケーションの双方に対応可能であるという点が挙げられる [11]。インスタントメッセージは本来同期的なコミュニケーションを行うために開発されたが、コミュニケーションに用いられるメディアが音声や映像ではなく文字情報であるため、録音や録画といった特別な操作をすることなしに、長時間に渡ってメッセージが保持されるという特徴がある。そのため、メッセージ受信時にユーザが端末の前を離れて

いたり取り込んでいて応答出来ないという言うような状況であっても、ユーザは自分の都合がついたときにメッセージを送ってきたユーザに返信し、その時点から改めて同期的なコミュニケーションを始めるといことが可能となる。

また、上記の特徴とも関連する事柄であるが、インスタントメッセンジャーによるコミュニケーションは何か他の作業を行いながら同時に行うことが容易であるという特徴を持つ [4, 11, 15, 16]。テキストによるコミュニケーションは音声や映像を伴うコミュニケーションと比較した場合、ユーザの集中力を奪う割合が小さい。すなわち、あるユーザが複数のタスクを同時進行的に行っている場合、それらのタスクに等しく労力が払われると言うことは少なく、ひとつのメインのタスクとその他のバックグラウンドで行われるタスクというように、ひとつのタスクがメインとなることが一般的である。このような状況において、音声や映像によるコミュニケーションを行う場合は必ずそのコミュニケーション自体がメインのタスクとなり、ユーザが行う自身の作業はバックグラウンドでのタスクとなってしまう。しかし、テキストによるコミュニケーションではコミュニケーション自体はバックグラウンドで行い、ユーザは自身の作業をメインのタスクとして行うことが容易である。これは、音声や映像が形として残らない媒体であるために常に注意を払っている必要があるのに対し、テキストでは文字という形でコミュニケーションの内容が示されるため、常にコミュニケーションそのものに注意を払う必要がないためであるといえる。

2.3.3 関連研究

本節では、これまで述べてきた CSCW やインスタントメッセージ、およびそれらにプレゼンスを用いることに関する関連研究について述べる。

2.3.2 節で述べたようなインスタントメッセージの持つ特性を利用し、これを拡張して地理的に離れた場所に存在しているユーザ同士のコミュニケーション、中でも特にフェイストゥフェイスの環境においてはごく自然に行われるインフォーマルなコミュニケーションを支援する試みがいくつかなされている。Mei Chuah は他のユーザからはメッセンジャーの 1 ユーザとして見えるプログラム (bot) を用意し、BuddyList 内で同一のストリーミング放送を視聴しているユーザの情報などを提供する試み [4] をおこなっている。ストリーミング放送を見るという行為自体の日常生活における占める割合がそもそも非常に少ないという点が難点であるが、これによって「リビングで一緒に同じテレビを見ているような感覚」が得られるとしている。

3° [32] では、再生中の曲をユーザ同士で共有して一緒に同じ音楽を聴くということを試みている。この「同じものを聴いている」ということにより同じ部屋で一緒に音楽を聴いているような感覚をユーザに与え、いわゆるユーザ同士の「繋がっている感」を得る事を目指している。ユーザが視聴するものが音楽か映像かという違いはあるが、コンセプトとしては Mei Chuah のアプローチ [4] に近く、どちらも「共有感」のコミュニケーションにおける役割について示す一例といえる。

またこれとは多少異なるアプローチとして、ConChat [24] ではユーザの様々なコンテキスト要素

を定義し、システムがユーザのコンテキストをルールベースで解釈しユーザのコミュニケーションを手助けすることを試みている。しかし、コミュニケーションがなされること自体を前提条件とし、行われているコミュニケーションをいかに豊かなものにするかということを目的としており、コミュニケーションを生起させることにもプレゼンスを活用する本研究とは着眼点が異なっている。また、全てのコンテキストに対応するルールを記述することは難しいことも考えられる。

また、プレゼンスという概念は、各々のユーザ（人間）についての様々な情報であることが一般的であるが、Cooltown プロジェクト [5, 18] においては対象を人間についてのみから全てのモノ（オブジェクト）に拡張し、それらのオブジェクトが全てそれぞれのプレゼンスを表記するウェブページを持つ。そして、各ユーザは適切に管理・設定されたアクセス権のもと、それらのプレゼンス情報にアクセスすることにより、ユーザが様々なオブジェクトについての情報を得るというものである。全てのオブジェクトがウェブ表記を持つことの実現性・妥当性についてと共に、そのような環境での膨大な情報量进行处理することの実現可能性についても検討を行う必要がある。

森川らはユーザ周辺の情報完了に関する情報などをユーザ状況に応じて集約し、体系化するサービスプラットフォームについて検討を行っている [34]。情報をユーザ自身の関わるものとユーザ周辺環境を表すものに分類するという点は本研究でのプレゼンス情報の扱いと同様の手法であるといえるが、その目的はコンテキストウェアサービスの提供のためであり、プロフィール情報の体系化に重点を置いている点が本研究とは異なっている。

また、発信者・受信者双方に最適なコミュニケーション手段を提供する試み [35] もなされているが、プレゼンス情報は主にシステムがユーザに最適なコミュニケーション手段を判断することに用いられており、プレゼンスによってコミュニケーションを活性化させるという本研究とは立場が異なる。また、選択肢としてシステムが用意したコミュニケーション手段しか提供されず拡張性という観点から問題がある。

2.4 おわりに

本章では、プレゼンス情報とはいかなるものであるか説明すると共に、それがコミュニケーションにおいてどのような役割を果たすかについて述べた。また、プレゼンス情報をコミュニケーションに活用する例として CSCW について述べると共に、CSCW などにおいて昨今その特性が見直されているインスタントメッセージについて説明し、その特徴を述べた。

第3章 インフォーマルコミュニケーションと プレゼンス

3.1 はじめに

第2章で見てきたように、地理的に離れた場所に位置するユーザ同士の円滑なコミュニケーションを支援するためにはプレゼンス情報の活用が不可欠である。しかし、一言でプレゼンスを活用すると言っても、どのような情報をプレゼンスとして他のユーザに広告するのか、具体的にどのようにしてプレゼンス情報を伝達するのか、またコミュニケーションを支援するとはそもそもどのようなことを意味するかなど、検討すべき課題は多い。

そこで、本章ではインフォーマルなコミュニケーションを支援するシステムを構築する際に留意すべき事柄について検討を行う。3.2節ではコミュニケーションを支援するということについて、コミュニケーションの質と量といった観点から検討を加える。3.3節では、プレゼンス情報が何に由来しているかという観点から個人に由来しているものと環境に由来しているものに分類することの妥当性について述べる。3.4節ではプレゼンス情報の流通について議論し、3.5節ではプレゼンス情報をユーザに呈示する場合に最終的にユーザが接することになるユーザインタフェースについて述べる。

3.2 コミュニケーションの内容と回数

本研究での目的を実現するようなシステムに於いては、なされるコミュニケーションはユーザの利益となると共に、ユーザにとって有用であるようなものでなくてはならない。集中して考え事をしているのにくだらないことで話しかけられたり、相手に「今どこにいるのか」ということのみを聞くために行われるコミュニケーションは、いくらコミュニケーションの回数が増えたところでユーザにとって有用ではない。その一方で、自分の作業が一息ついてちょっと雑談したいと思っているユーザが同士がコミュニケーションを行うことが出来れば、そのようなコミュニケーションは双方のユーザにとって有用であると推測出来る。上記の例のように同様の状況に置かれていたり、あるいは同じ事柄に興味を持つユーザ同士に対し、互いにその存在を知らせることでその事柄についてユーザにとって有益なコミュニケーションが行えることが期待出来る。ユーザの興味を軸にユーザ同士を結びつけること有用性は、Ping や Trackback を備えた blog サービスの昨今の隆盛からも見て取れる。

このように、ユーザにとっての有益性を上げることが最終的な目標となるが、その事とコミュニケーションの回数や頻度を増やすと言うことは必ずしも一致しない。第1章で述べたように、協調作業に不可欠なユーザ同士の信頼関係を築き上げるにはインフォーマルなコミュニケーションが重要であると考えられるが、必ずしもコミュニケーションを増やせば増やすほどユーザにとって有益性が増すとは言えない。

話しかけられる立場からすると、一人で集中して考え事をしているときに話しかけられて嬉しい人はいないだろうし、話しかける立場からすると、「誰でもいいから本郷の研究室にいる人を探している」という場合に、各ユーザの位置情報が不明な場合には片っ端から「今どこにいるか」ということを聞くことになり、聞く人・聞かれる人双方が不利益を被ることになる。一方で、自分の作業が一

息ついてちょっと雑談したいと思っているユーザが複数いる場合に、そのようなユーザ同士がお互いの状況を知ることが出来ればコミュニケーションを行う可能性が高まり、ユーザにとってより有益であるといえる。

以上のように、ユーザのコミュニケーションを支援する際にはコミュニケーションの回数は多いほどよいということは必ずしも言えず、またコミュニケーションの内容によってはそのコミュニケーションを行ったことでユーザが不利益を被る場合もあるということに留意する必要がある。しかし、適切なプレゼンス情報を活用することで無用なコミュニケーションを減らすと共に、ユーザにとって有益なコミュニケーションを活性化させることが期待出来る。ゆえに、本研究で検討するシステムにおいてもプレゼンスを活用し、インフォーマルコミュニケーションをユーザにとって有益なものとするための支援を行うものとする。

3.3 個人のプレゼンスと環境のプレゼンス

ある一人のユーザに注目した場合、そのユーザに関するプレゼンス情報は、一般にユーザ自身に由来するものとユーザを取り巻く環境に由来するものの 2 つに大別される [34]。ユーザ自身に由来するものとしてはユーザの位置情報、忙しさ、何をしているか、等といったものが挙げられる。一方、環境に由来するものとしては明るさ、温度、騒音レベルといったものが挙げられる。前者と後者を比較した場合、後者の特徴として実世界において比較的近距离に存在するユーザ同士は、同一かもしくはそれに準じた情報をプレゼンスとして保持するという点が挙げられる。

本研究で扱うシステムは屋内で用いられることを想定しており、その際には前段で述べた「環境」は概ね「部屋」に置き換えて扱うことができる。部屋に由来するプレゼンスとしては、前述の明るさなどの他に、張りつめている / 落ち着いている等といった部屋の雰囲気といったものも考えられ、この場合部屋の雰囲気を生じさせているのはその部屋に在室している人間である。このようにプレゼンスの種類によってはユーザ自身に由来するものと周囲の環境に由来するものが密接に関連する場合もあるため、本研究で検討するシステムにおいても両者とも扱える枠組みを持つことを目指す。

3.4 プレゼンス情報の流通

プレゼンス情報を利用する場合には、第 1 章で述べたように情報量を適度に制限することが重要となる。ユーザの状況を詳しく伝えるという観点からは、伝えるプレゼンス情報は粒度が細かい方が望ましいが、あまりに細かいプレゼンス情報は発信者の行動や置かれた状況というものを丸裸にしてしまう。「オフィスにおいて勤務時間中の社員の様子を上司が効率的に管理する」という目的の場合などにおいては、このようなユーザのプライバシーが大きく制限されるシステムが許容されるといえる。本研究で目的としているような同僚同士のコミュニケーションを支援するようなシステムの場合には、ユーザのプライバシーは最大限尊重されるべきである。また、そのよ

うにユーザのプライバシーを十分に考慮したシステムでなくてはユーザは気軽にシステムを利用することができず、「日常のインフォーマルなコミュニケーションを支援する」というシステムの目的を果たすことも困難になる。

そこで、発信者の周囲の実空間に存在する膨大な情報を、適度な情報量に制限して離れた場所の受信者にネットワークを介して伝達することを考える。その際、元の情報を削減して情報の粒度を落とす場所については以下のような3つの方法が考えられる。

- 情報の取得にカメラなどのような余り高機能なデバイスを用いず低機能なセンサなどを用いる場合など、実空間情報を取得する時点で情報を削減する方法
- ネットワーク上にサーバを置いて各ユーザに関する情報を全てサーバ蓄え、そのサーバ上で情報の粒度などを調整するというように、発信者から受信者へと情報を伝えるネットワーク上で情報を削減する方法
- ユーザに関する情報は全てユーザ自身の管理下に置いておき、相手ユーザに応じて粒度を調整して情報を送信すると共に、受信者側では各ユーザから送られてきた情報を必要に応じて合成し、ユーザに呈示する方法

これら3つの手法の比較検討を以下で行う。1つ目の手法ではそもそも発信者についての高度に詳細な情報を取得しないので、ユーザへの心理的な安心感が高いと言える。しかしその一方で、情報を加工したり複数の種類の情報を組み合わせてユーザに関する情報を作成するといったようなことを考えた場合には、実際に受信者に提示するより多くの情報を取得したいということも考えられ、そのような場合には2つ目や3つ目の手法の方が柔軟性が高くふさわしいといえる。また、2つ目の手法の場合はサーバを管理する管理者が必要になる。一般に管理者はサーバ上の全ての情報にアクセスできるため、ユーザが自分のプレゼンス情報をそのようなサーバに置くことに対して抵抗を覚える可能性も否定できない。一方3つ目の手法は、上記の2つの手法と比較してユーザのシステムに対する安心感と、プレゼンス情報の処理に対する柔軟性のバランスがとれていると言える。

以上のような観点から、プレゼンス情報の流通に関しては各ユーザが自身の情報を管理し、必要に応じて粒度を調整し他のユーザに公開するという手法を本研究では用いる。

3.5 ユーザとシステムの心理的距離

ユーザ同士のインフォーマルなコミュニケーションを促進させることを目標とするシステムにおいては、システムとユーザとの心理的な距離を適度に保つことが重要であると考えられる。ユーザが自分の作業に集中しているときでも、「口やかましく」他のユーザの様子を報告してくるようなシステムは非常に鬱陶しいものであるし、逆にふとした瞬間に他のユーザの様子をうかがおうと思ったときに、ユーザが多くの労力を割かなくてはならないようなものでは、そもそもシステム自体をあまり使おうという気分になれない。

以上は主に、ユーザに対して情報を提示する出力側のユーザインタフェース (UI: User Interface) についてであるが、入力側の UI についても同様のことが言える。システムが提供するプレゼンスの種類を増やす事にのみ注力し、その結果としてユーザが手動で入力しなくてはならない項目が膨大になってしまえば意味が無く、自動で取得できる情報は自動で取得してユーザへの負担を少なくしなくてはならない。

ユーザが自発的に使いたいと考え、かつ実際に継続的に使用するようなシステムであるためには、「ユーザの使用コスト < システムより得られるメリット」でなくてはならない。そのためには、システムが提供する機能を豊富にし、ユーザが享受するメリットを増やすことも勿論重要であるが、それと同時にユーザがシステムを使用するために必要とするコストを下げることも重要である。そのためには、システムの使い勝手、中でも特に、ユーザとシステムを直接結びつける UI の使用感を良好なものにする必要がある。一般的に、プロトタイプとして実験的に実装されたシステムでは、UI の実装は本質でないとされることが多いが、本稿で述べるようなシステムにおいては UI もまたシステムを構成する大きな要因であり、その出来不出来がシステムの使用感を大きく作用するという点で重要な要素であるといえる。そこで、UI は適度にユーザとの距離を保ちつつ、適度に主張するという特性が求められると考える。

また、この他にユーザおよびその周囲の状態を取得するに際しては、なるべく自動で行うべきであると考えられる。以下では、身近なセンシングによる情報取得の例として、自動ドアと比較を行うことでその理由を考察する。

自動ドアと手動のドアを比較した場合、自動ドアの方が両手がふさがっていても通行できるという点でユーザにとって利点が大きいが、依然として手動のドアも広く使われている。これは勿論コストの問題もあるが、ドアの場合はたとえ手動でもドアを操作しないと目的とする空間へ到達できないという代替不可能な機能があり、またかつユーザにとっては目的の空間へと到達することが至上命題であり、それによって得られるメリットが非常に大であるからといえる。

一方、本稿で目的としているようなシステムが、例えば各ユーザがどの部屋にいるかを取得して他のユーザに提示したいと考えた場合、そのような情報は各ユーザが手動で登録するのではなく、なるべくシステムがセンサなどによって自動的に取得することが求められる。身近な例として、筆者の周囲では MSN Messenger [19] を用いているユーザが多くいるが、そのスクリーンネームを“小野

@会議室”や“小野@25 教室”のようにすることで自分の位置情報を広告することは、比較的多くのユーザにより行われている。しかしこのようなスクリーンネームの変更はユーザが手動で行っているため、ユーザが考え事をしていたり、“うっかり”していた場合にスクリーンネームの変更を忘れるということがしばしば見受けられる。スクリーンネームによる位置情報の広告により、例えば会議室にいるユーザが「ちょっと頼みたいことがあるけど、今研究室にいるか？」と話しかけられるといった、いわば無駄なコミュニケーションを避けられるというメリットがあるが、このようなメリットはドアの例の場合ほど直接的でないため、上記のように位置情報の変更を忘れてしまうといえる。

このような観点から、筆者は本研究のようなシステムにおいては、一般に本質でないと思われることが多いユーザインタフェースについてもシステムを構成する重要な要素であり、ユーザビリティについても十分に考慮する必要があると考える。

3.6 おわりに

本章では、プレゼンス情報を用いてインフォーマルなコミュニケーションを支援する際に考慮すべき事柄について、コミュニケーションの質と量、プレゼンスが個人から由来するか環境から由来するか生起場所の違い、プレゼンス情報をいかにして流通させるか、ユーザとシステムを結びつけるユーザインタフェースはどのようなべきか、といった観点から検討すると共に本研究の取り組みの方向性を示した。

第4章 ACDCシステム

4.1 はじめに

本章では、第 3 章で検討した項目を反映させたテストベッドシステムとして本研究で提案する、インフォーマルコミュニケーション支援システム ACDC (Atmosphere Carrier for Daily Communication) のシステムデザインについて述べる。まず、4.2 節で本研究で提案するシステム構成の概要について述べ、4.3 節でユーザの周囲に様々な形で存在する情報をプレゼンス情報として生成する方法について述べる。そして、4.4 節で得られた情報をユーザ間で伝達する方法について述べ、4.5 節で実空間のユーザとネットワーク空間を結びつけるユーザエージェントについて述べる。

4.2 ACDC システム構成の概要

本節では、本研究で提案する ACDC の概要について述べる。ACDC システムの目的は、ユーザのプレゼンス情報を常時ユーザ間でやりとりすることでユーザ同士が「繋がっている感」を得ることを可能にするとともに、フェイストゥーフェイスの環境であったならばコミュニケーションが行われてしかるべき状況でも、ユーザ同士が地理的に離れた空間に位置しているためにその機会が失われているという事態を少しでも減らすことを目的とする。

このような目的を果たすためには、以下のような項目の実現が必要となる。

- ユーザの周囲の様々な情報を収集して解釈し、プレゼンス情報を生成すること
- 生成されたプレゼンス情報を、ユーザの望む相手にふさわしい粒度で通知出来ること
- 周囲のユーザから通知され受け取ったプレゼンス情報に対し、それぞれのユーザが自身の立場での解釈を加えることが出来ること
- 現実世界のユーザの存在をネットワーク空間に投影し、ネットワーク空間上でそれぞれのユーザを結びつけることが出来ること
- システムを常時使い続けることがユーザにとって苦にならないようなユーザインタフェースを備えていること

ACDC システムでは、各ユーザが占有する計算機上でソフトウェアを起動して使用することを念頭に置いている。また、職場や研究室などの同僚同士が、異なる部屋やあるいは遠く離れた拠点などフェイストゥーフェイスでコミュニケーションをとれない環境にいる場合でのインフォーマルなコミュニケーションの支援をターゲットとしており、基本的にユーザが職場や研究室などにいる場合にのみ使用することを想定しているため、屋外などをユーザが移動中に使用することは現段階では想定していない。

このように、現実世界では地理的制約によりコミュニケーションをとることが難しいユーザ同士を図 4.2 に示すようにネットワーク空間において結びつけ、フェイストゥーフェイスのコミュニケー

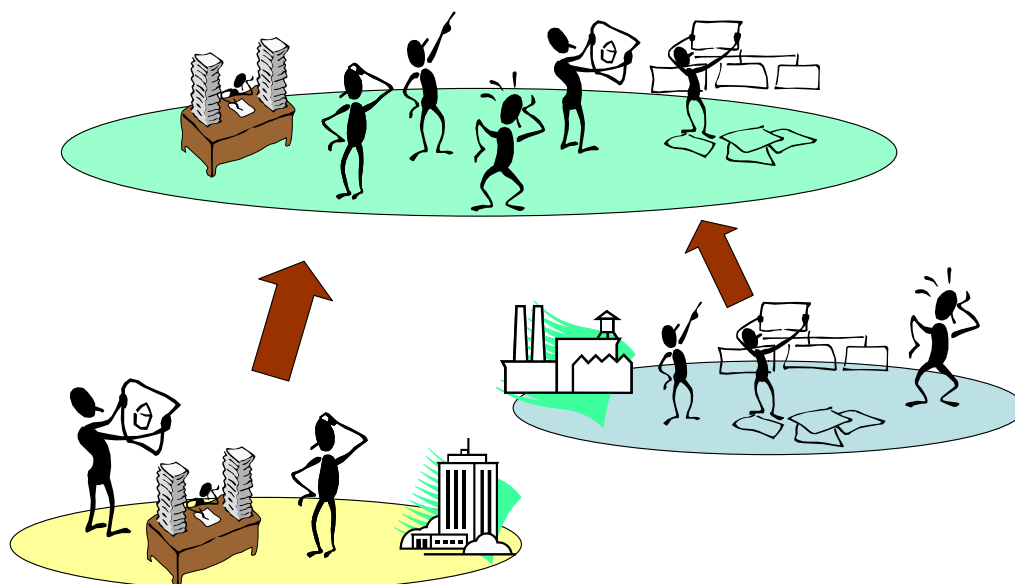


図 4.1: ネットワーク空間への投影

ションが行える環境と同等のコミュニケーションの提供を目指すというのが ACDC システムの最終的な目標である。

このような目標に対し、将来ユビキタス情報社会 [29] が実現した暁には、様々な情報が電子化されると共に、ユーザはセンサ類をはじめとして数多くのデバイスを利用することができるようになる。それらのデバイス類は種類も管理者も様々であるが機能から分別すると、図 4.2 に示すようにプレゼンス情報の対象である所有者、情報を要求し受信する受信者の他に、情報源（キーボード、センサなど）、情報提示装置（ディスプレイ、アクチュエータなど）が存在する。そして、これらと連携し、計算処理・データベース（DB）・通信の各機能を備え、実空間に存在しているユーザをネットワーク空間に投影するために存在するのがユーザエージェント（UA: User Agent）である。ユーザエージェントの機能については、4.5 節で詳細に述べる。

しかし、プレゼンス情報は極めて多岐にわたり、これらを全てユーザが利用出来るようにすることは現実問題として不可能である。ゆえに、どのようなプレゼンスを利用するかという取捨選択を行うことが重要になる。究極的にはある 1 種のプレゼンスを伝えることで本稿で目標としているようなコミュニケーションの支援が出来ることが理想であり、筆者はそのプレゼンスがどのようなものであるかについて検討を行っているが、現時点ではその解を見いだすに至っていない。さらなる検討を有意義なものとするためには、実際にプレゼンスを用いたインフォーマルなコミュニケーションの

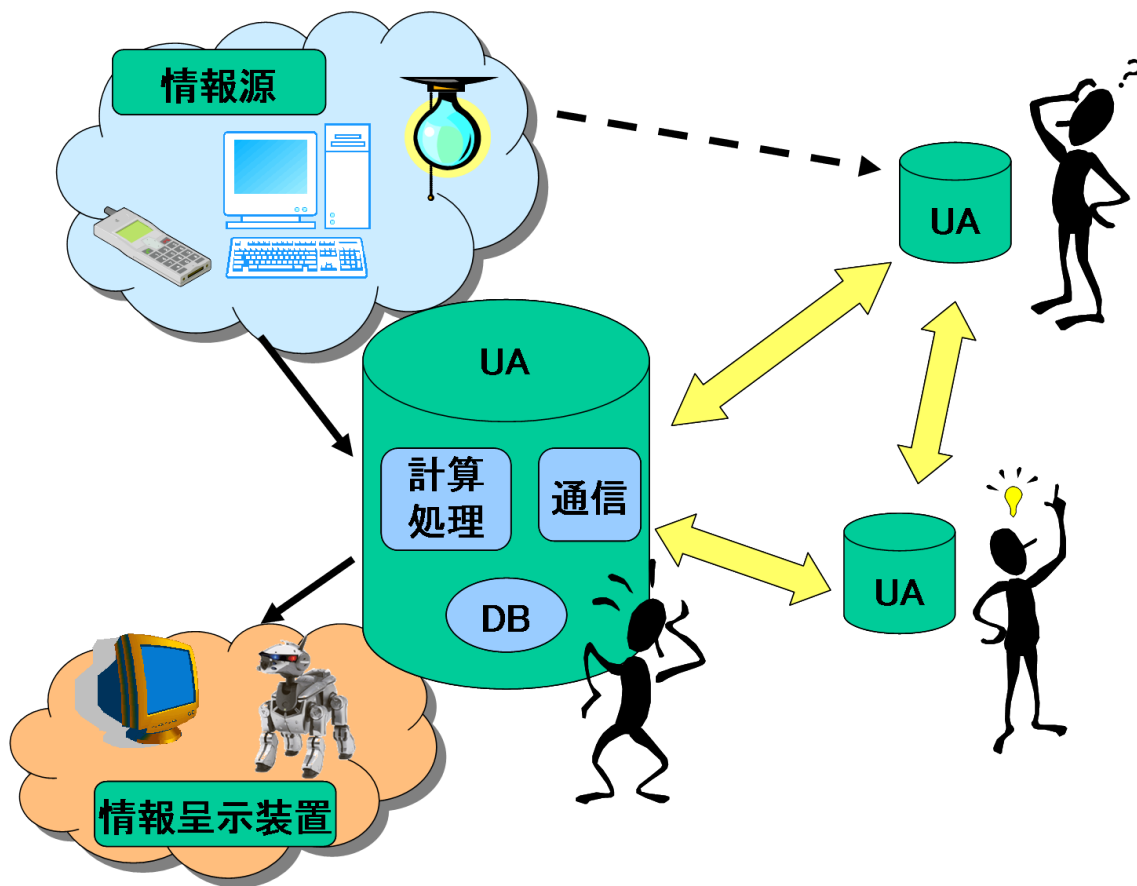


図 4.2: システム概要

支援システムを構築して試用していく中で、問題点の洗い出しを行うことが重要である。このような観点から、前章までで述べてきた検討事項を反映させ、課題の抽出を行うためのテストベッドアプリケーションとして ACDC システムは位置づけられる。

4.3 情報の取得

実空間に存在するユーザの状態をプレゼンスとして周囲のユーザに広告するためには、ユーザのコンテキストをコンピュータが扱える形に電子化し、ネットワーク上に投影する機構が必要となる。本節ではその枠組みについて述べる。

4.2 節で述べたような環境の中でユーザ（所有者）のプレゼンス情報を作成・通知する際には、ユーザの周囲に存在するセンサを利用して大量かつ高度なコンテキストを利用することや [5, 18]、各受信者に対して個別に情報公開ポリシーを設定し、各受信者との関係に応じた粒度のプレゼンス情報を公開できるよう、所有者を中心に情報源と保存者が連携して所有者に関する高度な情報生成が行われることが必要となる。また、情報源となるデバイスも各ユーザが占有して用いるものや複数ユーザに共有されるものなど様々な形態があり、それらに対応できる枠組みである必要がある。

4.4 情報の伝達と合成

各ユーザ毎に生成されたプレゼンス情報は、他のユーザへと広告されてはじめて意味を持つ。その際には、所有者は各ユーザ毎に異なる公開ポリシーを持つので、所有者と受信者が一対一で結びつけられて情報の伝達を行えることが必要となる。そのためには、各ユーザが一意的識別子を保持し、その識別子を用いて情報の伝達を行うことが必要となる。

また、一般に各ユーザの交遊関係は異なり、各所有者はそれぞれ異なったユーザリスト（BuddyList）を持つ上に、それぞれの相手に対して様々に異なる公開ポリシーを設定することが考えられるため、各ユーザが受信者として受け取る情報はユーザ毎に異なっている。さらに、同一の部屋などにいることがわかっている複数のユーザのプレゼンス情報を組み合わせることで、それらのユーザがいる環境に関するプレゼンス情報を取得することもできる。そこで、図 4.3 に示すように、他のユーザから受け取ったプレゼンスをそれぞれの受信者が各々解釈・合成を行うことで、各受信者の立場でプレゼンスを新たに創成しユーザに呈示できることが期待できる。

4.5 ユーザエージェント

前節までで述べたように、システムには

- プレゼンス情報は所有者毎に多様な情報源から集めて保存管理されること

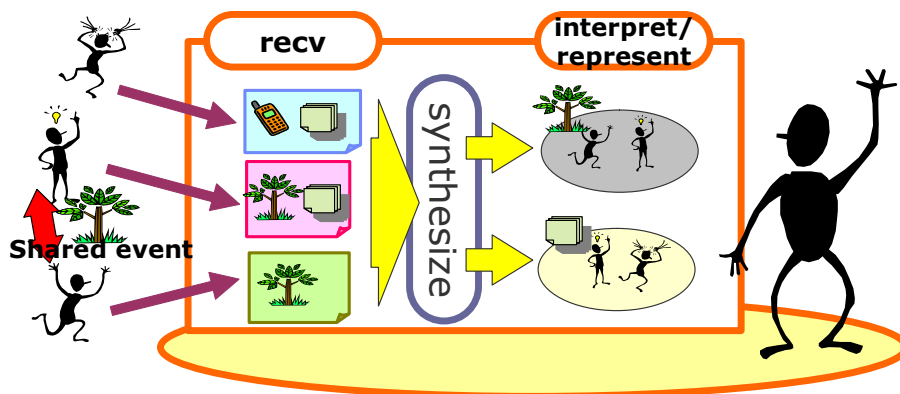


図 4.3: 受信者側での情報合成

- 所有者と受信者の関係を反映して公開する情報を制御できること
- 複数人から受信した情報を合成解析して付加情報やサービストリガーを生成する処理ができること

という条件が求められる。これらの条件を満たすと共に実空間のユーザの存在をネットワーク空間に投影し、ユーザが利用しやすい形で提供するために、情報の収集保存機能、相手毎にフィルタされた情報を伝える通信機能、合成等の受信情報の処理機能、およびユーザインタフェースやAPI (Application Program Interface) を併せ持つことがユーザエージェントに求められる。

図 4.4 にユーザエージェントの構成図を示す。前述した主要な3つの機能は相互に密に連携処理を行う。また、相手にプレゼンスを通知する際は必ず通信部を通す必要があるが、これらの要素はユーザインタフェースを含め同一端末にある必要はなく、例えばユーザインタフェースと通信機能は携帯電話を利用し、保存機能・処理機能は遠隔サーバで実行させるというシナリオも考えられる。ただし1ユーザ1エージェントに対応し、例えば複数のユーザに共同で利用されるデータベースサーバを置く場合であっても、あるユーザの保存部は他のユーザの保存部に直接アクセスできず、各ユーザ毎に独自に管理される。

4.5.1 通信部 (Network Transaction)

通信部 (図 4.4 下) は他のユーザエージェントとのプレゼンスや制御メッセージ通信を担う。そのためには、互いのユーザエージェントの通信識別子を使った接続処理から始まり、制御メッセージやプレゼンス情報そのもの、およびインスタントメッセージ (IM: Instant Message) のテキストデータなどの伝送処理、切断処理までを行う必要がある。

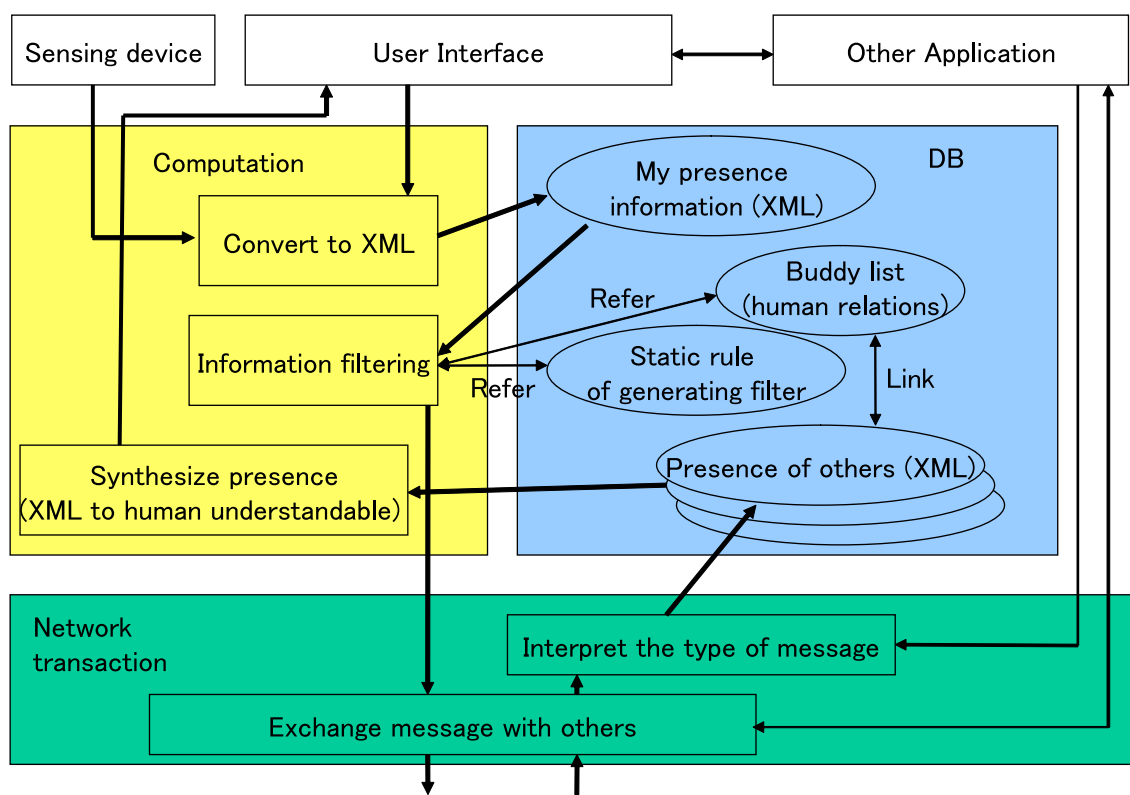


図 4.4: ユーザエージェントの構成

これらのプロトコルはアプリ独自でも良いが、SIP [25] がセッションの制御プロトコルとしてデファクトスタンダードとなりつつあり、SIP アドレスを識別子にして接続・切断処理を行い、メッセージのペイロードに情報を載せて伝達することができる。

SIP は IETF により RFC として標準化がなされておりこれに準拠したライブラリなどが多数公開されているため、これらのライブラリを利用することによって実装を行う際の労力を軽減することができる。また、SIP パケットはヘッダーと本文からなっているが標準として定められているのは基本的にヘッダーのみであり、本文にペイロードとして XML [7] などを格納することができ、拡張性に優れているという特徴がある。

具体的には、SIP をシステムのシグナリングプロトコルとして用い、ペイロードにプレゼンス情報や制御メッセージを載せ、受信時に送信元の確認、メッセージの種類の判定を行って保存部やアプリにデータを渡すという手法が考えられる。

4.5.2 保存部 (DB)

保存部 (図 4.4 右上) は、ユーザ自身および受信した他ユーザのプレゼンス情報を保存する。自身に関する情報は、情報源から集めた後にデータ処理を施したプレゼンス情報、情報送信先リストである BuddyList、および公開ポリシーを情報に適用するためのフィルタ規則を含む。また、プレゼンス情報は同一属性でも多様な表記方法があるため (例: 位置などの粒度、相対・絶対の種類別など)、それらを基本的に全て保存し、送信先との公開ポリシーに応じて情報を制御し、必要に応じてフィルタリングされた情報を生成することとする。データ保存形式は、抽出やマージ、データ変換などの管理・計算の容易性といった観点から XML を用いるものとする。

保存されているデータは、基本的にそれぞれの情報がアップデートされた時点で新しいデータに置き換えられ、古いデータは破棄される。ただし、複数の時刻のプレゼンス情報を合成することによって新たなプレゼンスを創成するという場合も考えられ、このような要求がある場合には、図 4.5 に示すようにイベントの生起によりプレゼンス情報が更新されてから次にイベントが生起してプレゼンスが更新されるまでを 1 世代として、指定された世代の分の過去のプレゼンス情報についても時系列別に保存しておくものとする。

4.5.3 処理部 (Computation)

処理部 (図 4.4 左上) は、各種のデータ変換・合成や情報のフィルタ処理を行う。

アプリケーションの起動時や、センサやユーザインタフェースなどの情報源のデータが変更された場合にはユーザの状況 (コンテキスト) を生成する処理を行って保存部へと渡すことを行う。なおこの際、センサやユーザインタフェースのような情報源のデータが変化したことを検出するのは、情報源自体がデータの変化を検出してイベントとして通知する場合と、ユーザエージェント側が定

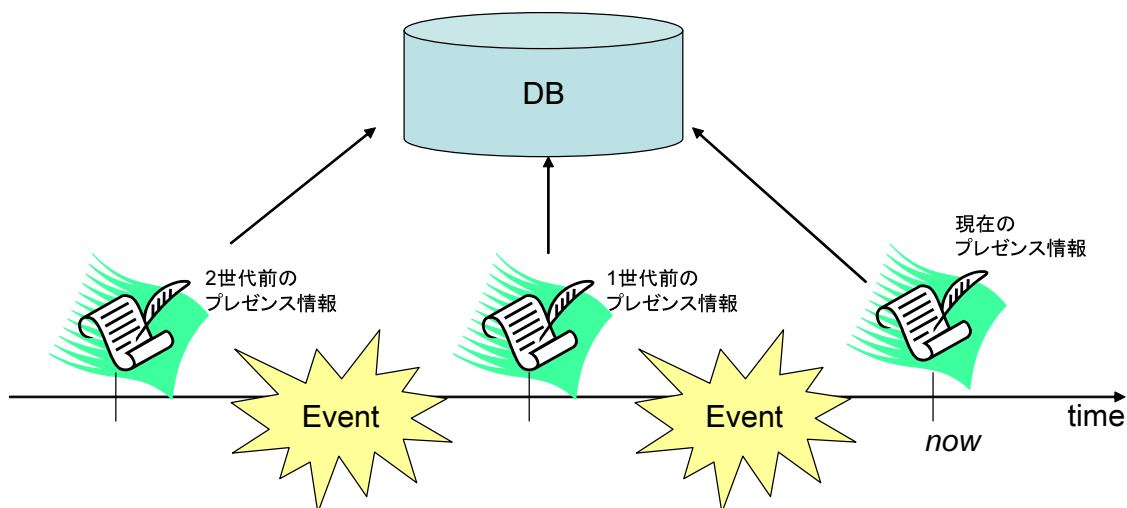


図 4.5: 過去のプレゼンスの DB への記録

期的に情報を取得して前回のデータとの比較を行い、データに変更が加えられたかどうかをチェックする方法（ポーリング）の2通りが考えられるが、そのどちらにも対応出来るものとする。

実際にプレゼンス情報を他のユーザに向けて発信する場合には、保存部に保存されているプレゼンス情報とフィルタ規則から他人に公開する自分自身のプレゼンス情報の生成処理を行う。発信のタイミングは、センサやユーザインタフェースといった情報源のデータが変化して自身のプレゼンス情報が変化した場合に自分から情報を発信するケースの他に、他のユーザエージェントから情報を通知することを要求され、それに応える形で自身のプレゼンス情報を送信するというケースが考えられる。前者の場合は、後者の場合と比較して BuddyList 内の多くのユーザに一斉に自身のプレゼンス情報を送信することになるため、そのような負荷に耐えられるだけの計算能力資源およびネットワーク資源をユーザエージェントは備える必要がある。

また、情報受信者側では、BuddyList 中の相手から受信したプレゼンス情報群を解析することで、それらのプレゼンス情報軍のなかに含まれる共通項目や、自身と相手ユーザとの関係から合成プレゼンスを生成し、ユーザやアプリケーションに提示可能な状態に変換する。変換されたデータは API を通じて他のアプリケーションに通知されたり、ユーザインタフェースを通じてユーザに呈示される。

4.6 おわりに

本章では、本研究で提案するインフォーマルなコミュニケーションの支援を目的とするシステム ACDC の構成について概要を 4.2 節で述べると共に、システムにおける情報の取得および伝達、プレゼンス情報を受信したユーザの立場におけるプレゼンス情報の合成について 4.3 節と 4.4 節で述べ、各々のユーザをネットワーク空間に投影するユーザエージェントの構成について 4.5 節で述べた。

第5章 コミュニケーションに与える影響の 評価

5.1 はじめに

本章では、本研究で述べているプレゼンスを用いてコミュニケーションの活性化をはかる事の妥当性の確認と、さらなる課題の抽出を行うために実装したアプリケーションについて説明を行うと共に、実装を行ったアプリケーションを用いてプレゼンス情報のコミュニケーションに与える影響を明らかにするために行った評価実験について述べる。

5.2 ACDC システムの実装

本節では、ACDC システムとして実装を行ったアプリケーションについての報告を行う。

5.2.1 実装方針

本実装では、現在各ユーザが占有している各々の PC 上でクライアントを動作させることを前提とし、Windows XP Service Pack 2 上で実装を行った。また、コミュニケーション機能とプレゼンス機能を兼ね備えたツールとして MSN [19] や ICQ [14] に代表されるメッセージャーが存在するが、本実装ではこのようなメッセージャー機能をベースとしつつ、独自のプレゼンス機能を搭載したアプリケーションとして実装した。

既に述べたように、プレゼンスの種類は極めて多岐にわたり、それらを全て実装することは実際問題として不可能であり、どの種のプレゼンスを実装するかという点が問題になる。究極的にはある1種のプレゼンスを伝えることで本稿で目標としているようなコミュニケーションの支援が出来ることが理想であり、筆者はそのプレゼンスがどのようなものであるかについて検討を行っているが、現時点ではその解を見いだすに至っていない。そこで、本実装では有用であると期待できるプレゼンス情報を数多くユーザに呈示することでコミュニケーションの支援を目指すものとした。具体的にどのようなプレゼンスを用いたかについては 5.2.3 節で述べる。

5.2.2 フレームワーク

前節で述べたような様々なプレゼンス情報をやりとりする際に用いる、アプリケーションの土台となるフレームワークについて述べる。

シグナリングプロトコルとしては 4.5 節で述べたように SIP を用い、各ユーザの識別子として SIP アドレスを用いた。また、SIP ライブラリとして GNU oSIP library [28] を用い、これを用いて作成した Win32 DLL を .NET Framework 上で動作する C# で記述したアプリケーションから呼び出す構造とした。そして、各ユーザのプレゼンス情報は SIP の NOTIFY メッセージのペイロードに格納され BuddyList 内のユーザに広告される。ネットワーク上には SIP サーバが設けてあるが、これはネット

ワークレベルでの各ユーザの SIP アドレスと IP アドレスの対応付けにのみ用いられており、プレゼンス情報レベルではサーバレスの構成としている。

また、プレゼンスの最終的なユーザへの呈示は基本的に PC のモニターによっているが、一部情報を AIBO [1] を通じて実空間へ反映させる事も試みている。図 5.1 に作成したアプリケーションのスクリーンショットを示す。左がツールチップによる情報表示の、右がデスクトップ上へのポップアップによる注意喚起の例である。スクリーンショットは BuddyList 内の全員を表示させる画面であるが、アプリケーションのメインウィンドウ左側のタブで各部屋別に在室しているユーザを表示させることもできる。

SIP のシグナリング

以下では、本実装で利用した SIP のシグナリングについて述べる。

まず、各ユーザがアプリケーションを操作し REGISTER メッセージを SIP サーバに送信することでサインインすると同時に、SIP の SUBSCRIBE メッセージが BuddyList 内の各ユーザの SIP アドレスに対して送付される。その時点でオフラインのユーザについては SIP サーバから 404 (Not Found) 応答がかえされる。一方、既にサインインしているユーザに対しては SIP サーバによりメッセージが転送される。SUBSCRIBE を受け取ったユーザエージェントは、自身のその時点のプレゼンス情報を格納した NOTIFY メッセージを送り返し、これによってサインインしたユーザは他のオンラインのユーザのプレゼンス情報を知ることができる。また、SUBSCRIBE メッセージにはヘッダに expires (有効期間) フィールドが定められており、これが有効である間に自身のプレゼンス情報が変化した場合には新たなプレゼンス情報を NOTIFY メッセージに格納して再び送信する。各々のユーザエージェントは、自身が送信した SUBSCRIBE メッセージの有効期限が切れる前に新たな SUBSCRIBE メッセージを送信し、有効期限が失効することを防ぐ。プレゼンス情報を受け取る側 (subscriber) と発信する側 (notifier) との間のパケットの流れを図 5.2 に示す。以上は SIP の標準に基づいた動作であるが、このような動作を行うことによってあるユーザのプレゼンスが変化した場合にその情報を即座に他のユーザに伝達することができる。同時に、アプリケーションエラーやネットワークの不調などで正常終了せずにユーザがシステムを利用出来なくなった場合でも、実際にはオフラインのユーザが他のユーザからオンラインに見えている期間は SUBSCRIBE メッセージのヘッダの expires が上限となる。expires の値を小さくすれば異常終了したユーザをすぐにオフラインとして認識出来るが、この値の大小と SUBSCRIBE メッセージのトラフィック量はトレードオフの関係にあるため適度な大きさにする必要がある。本実装では expires の値はユーザが指定出来るようにしたが、デフォルトでは 300 秒とした。

また、前述の通り本アプリケーションはインスタントメッセージングをベースとしており、テキストチャットによるコミュニケーションをサポートしている。SIP は当初セッションの制御目的で定め



図 5.1: スクリーンショット

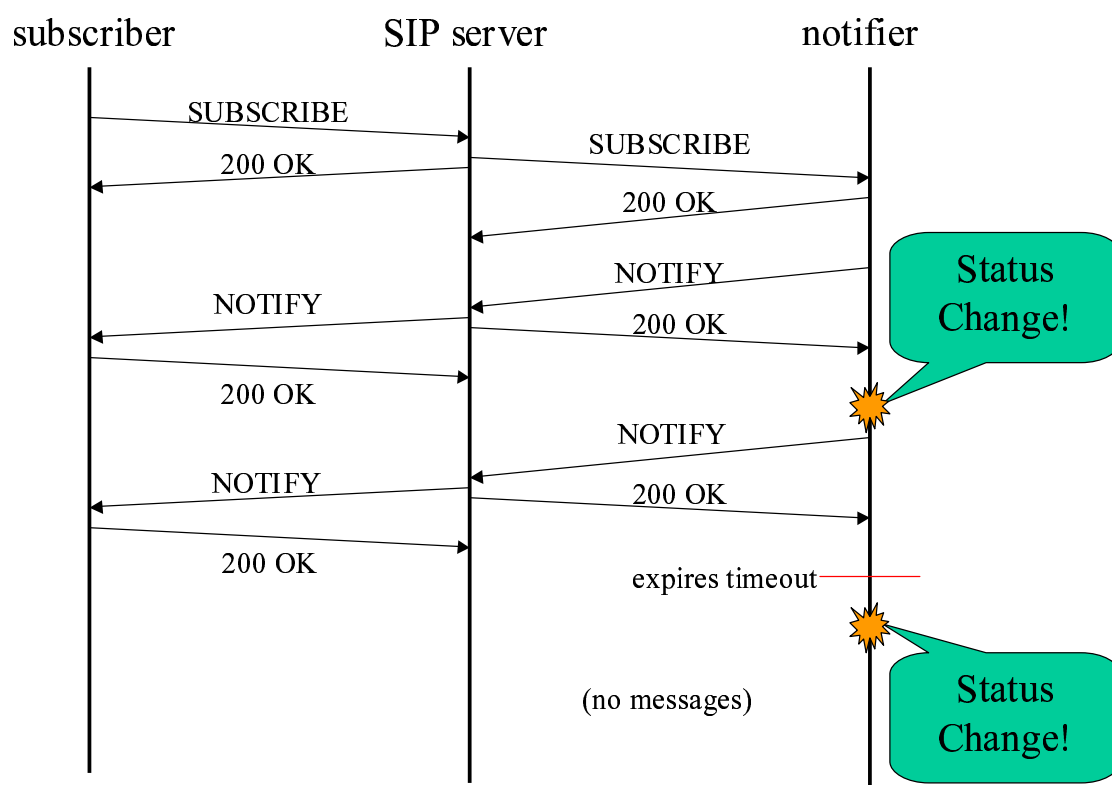


図 5.2: プレゼンス伝達のパケットの流れ

られたプロトコルであるが、今日ではインスタントメッセージの伝送についても RFC で正式に標準化されており [2]、本アプリケーションもこれに準拠したプロトコルを用いている。即ち、話しかける側 (caller) と話しかけられる側 (callee) との間で図 5.3 に示した手順によりセッションを確立し、このセッション内で MESSAGE メッセージをやりとりしてインスタントメッセージングを実現している。具体的にセッションを確立し、会話が行われ、セッションが終了するまでの手順は以下の通りである。

1. caller が callee 宛に INVITE メッセージを送る。メッセージはまず SIP サーバに届けられる。
2. メッセージを受け取った SIP サーバは caller に 100 (Trying) を応答すると共に、INVITE メッセージを callee に転送する。
3. INVITE メッセージを受け取った callee は 180 (Ringing) 応答を SIP サーバを通じて caller へ送る。
4. callee がセッションへの招待を受諾した場合には同様に 200 (OK) を caller へ応答する。VoIP (Voice over IP) などのアプリケーションの場合にはユーザによる確認があってはじめて 200 が返信されるが、本アプリケーションではインスタントメッセージングであるため、ユーザの操作を特に必要とすることなく自動的に 200 応答を返信する仕様としている。
5. 200 応答を受け取った caller は ACK メッセージを callee に送り、これによって caller と callee の間でセッションが確立する。
6. ユーザがインスタントメッセージの会話ウィンドウを閉じるなどの操作を行いセッションを終了する場合には、BYE メッセージを送信する。
7. BYE メッセージを受信した場合には 200 応答を返し、これによってセッションは終了する。

それぞれの SIP パケットがどのセッションに含まれるかは、SIP パケットのヘッダーに含まれる Call-ID フィールドの値による。Call-ID はセッションの識別子であり、これが同一のパケット同士がひとつのセッションに関する SIP パケットとして認識される。

なお、2人以上のユーザが同時に参加するチャット (マルチチャット) の場合には、すべての参加ユーザ間にフルメッシュで一対一のセッションを設けている。その際のそれぞれのセッションの Call-ID は全て同一である。

DLL の提供する API

本実装では、前述のように SIP を含めた通信機能を DLL 化して実装を行った。以下では、DLL がアプリケーション本体 (exe ファイル) 向けに提供する API について述べる。表 5.1 に API の一覧と戻り値を示す。以下でそれぞれの関数について詳細に述べる。

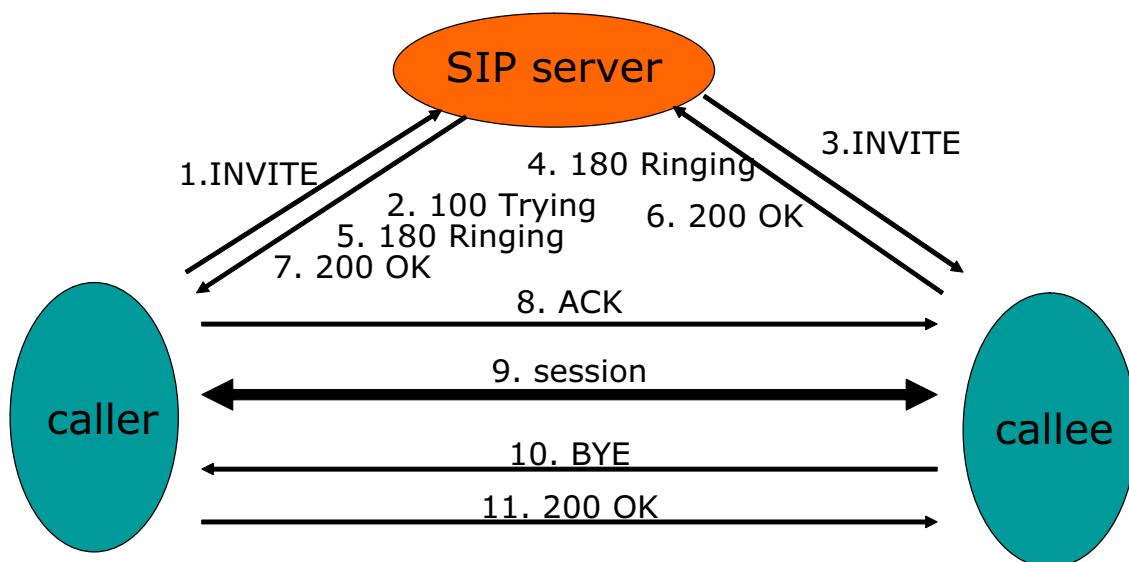


図 5.3: SIP によるチャットセッションの確立

`DllInit(char* ServerAddr, char* MySipAddr, int localport, int expires, char* MyIPAddr)`

DLL を初期化する関数であり、他の API を起動する前にこの関数を呼ぶ必要がある。引数は SIP サーバのアドレスである *ServerAddr*、自分の SIP アドレスを表す *MySipAddr*、SIP サーバとの通信に用いるポート番号を指定する *localport*、REGISTER パケットの有効期間を示す *expires*、SIP サーバとの通信に用いる自分の IP アドレスを指定する *MyIPAddr* である。

`SignIn(void)`

SIP サーバに対して REGISTER メッセージを送る。REGISTER メッセージを送ることによって、以後 SIP サーバはユーザの SIP アドレスと IP アドレスの対応付けを行うことが可能になる。なお、SIP サーバのアドレスなど通信に必要なパラメタ類は `DllInit()` によって指定されたものを用いる。

`FirstSignIn(char* RemoteSipAddr)`

SIP アドレスが *RemoteSipAddr* のユーザに対して、自分を SUBSCRIBE するよう要請する magic string を MESSAGE メッセージを用いて送付する。この magic string を受け取ったアプリケーションは、送付してきた相手に対し SUBSCRIBE メッセージを送り返す。

表 5.1: DLL の提供する API 一覧

戻り値	関数名及び引数
bool	DllInit (char* ServerAddr, char* MySipAddr, int localport, int expires, char* MyIPAddr)
bool	SignIn (void)
bool	FirstSignIn (char* RemoteSipAddr)
bool	SignOut (void)
bool	SendSubscribe (char* RemoteSipAddr, int expire)
bool	PresenceChange (char* Presence)
bool	BeginChat (char* RemoteSipAddr, char* CallId)
bool	FinishChat (char* CallId)
bool	InviteChat (char* RemoteSipAddr, char* CallId, char* Contents)
bool	JoinChat (char* RemoteSipAddr, char* CallId)
bool	SendChatMessage (char* CallId, char* contents)
void	GetEventCallBackFunc (CallBackFunc callBackFunc)

SignOut(void)

SIP サーバに、expires フィールドが 0 の REGISTER メッセージを送ることによって SIP サーバから自身のエントリを消去する。この関数が呼ばれることなくアプリケーションが終了した場合、最大で expires 秒の間実際には通信不可能になっているにもかかわらず SIP サーバ上にエントリが残ってしまうという不都合が起こる。

SendSubscribe(char* RemoteSipAddr, int expire)

RemoteSipAddr の相手に対して、ヘッダの expires フィールドを *expires* に指定した SUBSCRIBE メッセージを送信する。SUBSCRIBE メッセージを受信した側は直ちに NOTIFY メッセージに自分のプレゼンス情報を格納して返信する。この一連の動作によって相手のプレゼンス情報を得ることが出来る。

PresenceChange(char* Presence)

現時点で有効な SUBSCRIBE メッセージを送ってきている相手ユーザに対して、*Presence* の内容を NOTIFY メッセージに格納して送信する。“有効な SUBSCRIBE メッセージ”とは有効期限の切れていない SUBSCRIBE メッセージのことであり、具体的には SUBSCRIBE メッセージを受信してから、そのヘッダの expires フィールドの値の秒数以上経過していないメッセージのことである。

BeginChat(char* RemoteSipAddr, char* CallId)

RemoteSipAddr で指定される相手との間に、チャットを行うためのセッションを構築する。*CallId* はその際にランダムに決定され、exe 側へ通知される。以後 exe 側では、会話セッションは基本的にここで決定された *CallId* により識別・管理される。

FinishChat(char* CallId)

CallId で指定される全ての会話セッションに対し、BYE メッセージを送信して終了する。

InviteChat(char* RemoteSipAddr, char* CallId, char* Contents)

CallId で指定される既存の会話セッションに、新たに *RemoteSipAddr* のユーザを INVITE メッセージを送信することで招待する。なお、本実装での会話セッションは 1 対 1 のセッションを基本としており、複数人によるマルチチャットを行う際には、参加者全てに対しフルメッシュで同一 *CallId* のセッションをはる必要がある。そこで、新規参加者に既存の参加者を知らせるために、*Contents* として既存参加者の SIP アドレスのリストを受け取り、これを INVITE メッセージの本文に格納して送信する。この INVITE メッセージを受け取ったクライアントは、INVITE メッセージを送信してきたユーザとセッションを構築すると共に、本文に格納されたユーザ全てに対しても INVITE メッセージを送信してセッションを構築する。

JoinChat(char* RemoteSipAddr, char* CallId)

コミュニケーションプレゼンスにより、既存の会話に第 3 者から加わることを求める場合に用いる。なお、コミュニケーションプレゼンスについては 5.2.3 節で述べる。*RemoteSipAddr* で指定されるユーザに対して、*CallId* の *CallId* を持つ会話に参加することを求める。なおこの関数は、*RemoteSipAddr* を変更することで会話に参加している全てのユーザに対して用いられる。

SendChatMessage(char* CallId, char* contents)

CallId で指定される会話セッションに対し、*contents* の内容のテキストメッセージを送付する。会話が 3 人以上で行われている場合には、全てのユーザに対して *contents* の内容が送付される。

GetEventCallBackFunc(CallBackFunc callBackFunc)

DLL 側から exe 側へイベントを通知するために用いられる。この関数を用いることで、コールバック用の関数のアドレスが exe 側へ通知される。コールバック関数はイベントの種類を指定する int 型の引数を 1 つと、イベントに応じて意味が変わる char* 型の引数を 3 つ持つ。イベントの種類と char* 型の 3 つの引数の対応を表 5.2 に示す。以下でそれぞれのイベントについて述べる。

PRESENCE_CHANGE : *RemoteSipAddr* のユーザから、*Presence* というプレゼンス情報が含まれた NOTIFY メッセージを受信したことを示す。

表 5.2: コールバック関数のイベントと引数

イベント名	引数 1	引数 2	引数 3
PRESENCE_CHANGE	RemoteSipAddr	Presence	(unused)
CHAT_INVITED	RemoteSipAddr	CallId	Contents
MESSAGE_RECV	RemoteSipAddr	CallId	Contents
CHAT_FINISHED	RemoteSipAddr	CallId	(unused)
CHAT_NEW_JOINED	RemoteSipAddr	CallId	(unused)

CHAT_INVITED : *RemoteSipAddr* のユーザから , *CallId* という *CallId* の値をもつ *INVITE* メッセージが届いたことを示す . 1 対 1 の会話に誘われた場合は *Contents* の中身は空であるが , 既存のチャットに誘われた場合は既存参加者のリストが *Contents* に含まれる .

MESSAGE_RECV : *CallId* で指定される会話セッションに対し , 発言内容を *Contents* に含む *MESSAGE* メッセージが *RemoteSipAddr* のユーザから届いたことを示す .

CHAT_FINISHED : *CallId* で指定される会話セッションから , *RemoteSipAddr* のユーザがチャットウィンドウを閉じることで *BYE* メッセージを送り , 脱退したことを示す .

CHAT_NEW_JOINED : *CallId* で指定される会話セッションに対し , *RemoteSipAddr* のユーザが新規に参加したことを示す . 前述の *JoinChat()* による参加要請を受信した場合に対応する .

5.2.3 プレゼンス情報

本節では , 実際に実装を行ったプレゼンス情報の種類について , その概要とどのような意図に基づいてのものかについて述べる .

ステータス

自身の状況について , 「PC から離れている」「多忙につき話しかけるな」等と言ったことを意味する数種類のステータスから選びプレゼンスとして他のユーザに広告するものであり , 多くのインスタントメッセージングソフトにおいて実装されている極めて基本的なプレゼンス情報である . 本実装においては , 「オンライン」「取り込み中」「休憩中」「電話中」「退席中」の中から , ユーザが手でマウス操作により選択するものとした .

スクリーンネーム

自身の名前として相手ユーザの画面に表示されるものであり、前項と同様に多くのインスタントメッセージソフトが備えている機能である。本来、ユーザの同定は SIP アドレスなどのアカウントにより可能であるが、ユーザにとってアカウント名と実際の名前を結びつけることは難しいことがあるため、自身の名前をフリーテキストで入力することでそれが相手に表示されるというものがある。このように名前を入れることを意図して設けられたスクリーンネームであるが、一部のユーザのなかにはここに名前以外に近況報告のような短信を併記するユーザも見られる。

コミュニケーションプレゼンス

実空間においては誰と誰が会話をしているかという情報は周囲の人間が容易に知ることのできる情報であり、またこのような情報を蓄積することで、我々は人間関係を窺い知ることができる。そこでこれをネットワークコミュニケーションにも発展させ、ネットワークコミュニケーションで誰と誰が会話をしているかという情報をプレゼンスとして周囲に伝達させることを目的とするのが“コミュニケーションプレゼンス” [33] である。本実装ではコミュニケーションとして文字チャットを扱い、誰と誰がチャットを行っているかを周囲のユーザに広告する。チャットは 1 対 1 の通常のチャットのみでなく、3 人以上のユーザが参加するマルチチャットにも対応している。

この際実際にプレゼンス情報として伝達されるのは、会話のセッション ID と会話への参加人数だけであり、会話に参加している全てのユーザのプレゼンスを受け取れるユーザのみが、誰と誰が会話をしているかを実際に把握できる仕様としている。なお、会話のセッション ID としては SIP の Call-ID の値を用いている。ユーザに呈示する UI 画面は図 5.4 の様になっており、左がどのユーザも会話をしていない状態、右が一部のユーザが会話をしている状態である。図中の中心の点が自分を、周囲の点が他のユーザを表しており、会話をしているユーザは線で繋がって表示される。そして、ユーザが他のユーザ同士の会話を見てそれに参加したいと考えた場合には、会話を行っているユーザをクリックすることでその会話へ参加することが可能である。

部屋の雰囲気

部屋の「雰囲気」を測る尺度には様々なものが考えられるが、本実装ではそのうち部屋の空気がどの程度緊張しているかという軸に注目してこれをプレゼンスとして広告するシステムを実装した。ただし、「どの程度部屋が緊張しているか」というコンテキストを自動で取得することは極めて困難であるため、これをユーザが手動で入力するシステムとした。具体的には図 5.5 に示すように、各ユーザは自分の在室している部屋名と“緊張度”をプレゼンスとして周囲のユーザに広告する。一方で、周囲のユーザから受け取ったプレゼンス情報を解釈し、各部屋毎に在室しているユーザの申告する緊張度に対して何らかの処理 ($f(x)$) を行って部屋の緊張度を求め、ユーザに提示する。なお、

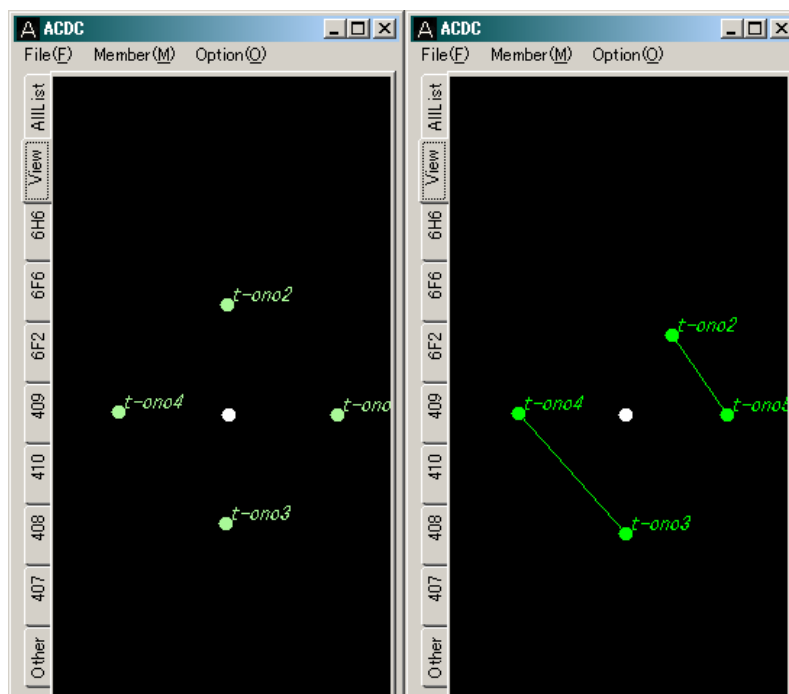


図 5.4: コミュニケーションプレゼンスの表示例

緊張度は周囲のユーザのプレゼンスが更新され新たなプレゼンスが届くたびに再計算される。そして、新たに求められた緊張度が以前より増加して閾値を超えた場合は、デスクトップの隅に図 5.1 右のようなポップアップを表示させユーザに注意を喚起する。

本実装においてはユーザが広告する「緊張度」は 0 から 100 までの整数値であり、数字が高いほど緊張度が高いことを示す。そして各部屋毎に、それぞれの部屋内のユーザが申告してきた値に対する処理 $f(x)$ として加算平均を用いて「部屋の緊張度」を求めている。

なお、本実装ではユーザがどの部屋にいるかという情報はユーザがプルダウンボックスから選択するようになっている（図 5.6）が、これを RFID 等を用いたユーザ位置検出システムなどと組み合わせることで自動化させることも考えられる。また、プルダウンの一覧にない部屋にユーザがいる場合は部屋名をフリーテキストで入力することもできる。ただしその場合には、自分の部屋の「緊張度」を他ユーザに伝えることは出来ない。

BGM 情報

ユーザ同士が共通の趣味を発見して話が弾む事を期待し、話題の「種」を提供することを目的として、ユーザが BGM として再生している音楽の曲情報を取得しプレゼンスとして周囲に広告するよ

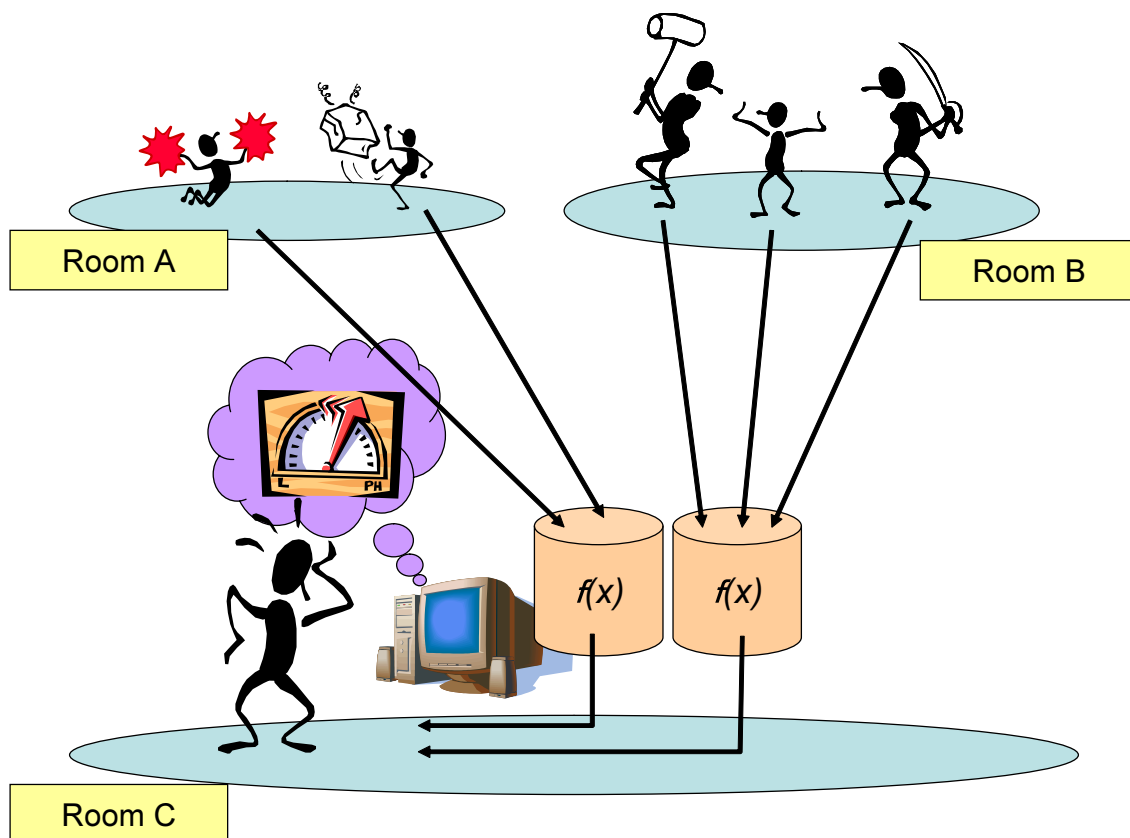


図 5.5: 部屋の雰囲気伝達

うにした．具体的には Winamp を COM サーバにするプラグイン [30] を用い，本アプリケーションとの間で COM による通信を行って再生中のメディアファイルに対して，タイトルとアーティスト名をファイルのメタ情報から取得している．最も一般的な MP3 ファイルの場合には，MP3 ファイルに ID3 タグが埋め込まれている場合はタグ内容を取得する．ID3 タグがない場合にはファイル名をそのままタイトルとしている．

他のユーザから受信した BGM 情報は，図 5.1 左のように BuddyList をマウスポインタでポイントしたときに表示されるツールチップを用いてユーザに表示するものとした．BuddyList の一覧画面の中で「」がついているユーザが現在音楽を聴いているユーザである．なお，図 5.1 に示すスクリーンショットでは Windows の仕様によりマウスポインタが表示されていないが，実際には選択されて反転表示されているユーザのあたりをポイントしている．この方法ではユーザがマウス操作を行わない限り情報が表示されず一覧性がよくないという欠点があるが，BuddyList の表示が煩雑になるのを防ぐことを優先しこの方法を採用した．また同時に，一定時間毎に BuddyList 内のユーザのうち



図 5.6: 自分の部屋の選択画面

音楽を聴いているユーザを一人ランダムで選択し、図 5.1 右のようなポップアップを表示させてそのユーザが聴いている曲情報を表示させることも行った。

会話回数

ユーザの忙しさとコミュニケーションの回数について考察すると、忙しいときはコミュニケーションの量が減り、暇なときはその逆と言うように何らかの相関があると期待される。そこで、アプリケーションを起動してからのインスタントメッセージのセッションの累積数をプレゼンスとして広告することとした。なおユーザへの表示は、前述の BGM 情報と同様に図 5.1 左のようなツールチップを用いた表示としている。なお、この回数はアプリケーションを一旦終了して起動し直すと 0 に戻るものとしているが、これ以外に「最近 3 時間の会話数」というように時間によって区切ってセッション数を数えるという方法も考えられる。

騒音レベル

ユーザの在室している部屋がうるさいか静かかというプレゼンスを伝達することで、部屋の「活気」をある程度伝達できるのではないかという意図のもとに部屋の騒音レベルを 3 段階で広告することにした。騒音レベルの取得は AIBO に搭載されている無指向性マイクを用い、騒音レベルは「AIBO のプレゼンス」としてユーザに広告される。ユーザは BuddyList の AIBO をマウスでポイントすることにより表示されるツールチップにより、その AIBO が置かれた部屋の騒音レベルを知ることができる。なお、AIBO についての詳細は 5.2.4 節で述べる。

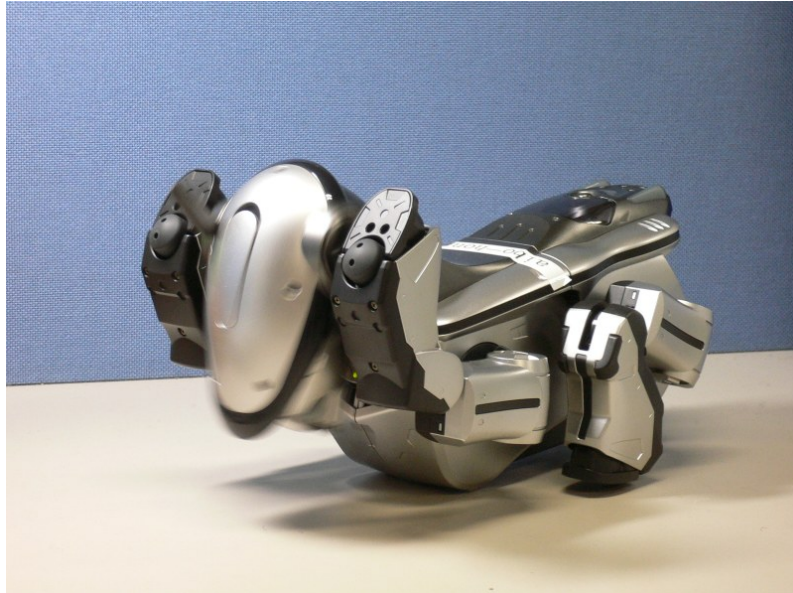


図 5.7: AIBO による情報呈示例

5.2.4 実空間との入出力連携

PC のモニタ以外を通じてプレゼンス空間と実空間をつなぐ試みとして、AIBO を用いて 5.2.3 節で述べた部屋の雰囲気を実空間に反映させることを行った。具体的には、それぞれサーバにより制御される AIBO を各部屋に一匹ずつ配置する。各々の AIBO は他のユーザからは一般ユーザと同様の 1 クライアントとして見える。各ユーザは AIBO を BuddyList に登録し、他のユーザへと同様に自分のプレゼンスを AIBO に通知する。そして AIBO は受信したプレゼンスを合成し、自分がいる場所以外の部屋の“緊張度”が上昇した場合に図 5.7「おびえた」仕種をする。これにより、ユーザの意識が PC 以外を向いている場合でもユーザに遠隔地の状況が変化したことを伝えると共に、AIBO の仕種による一種の「癒し」効果が期待できる。

また、これと同時に AIBO に内蔵された無指向性のマイクを用いて、AIBO が置かれた部屋の騒音レベルについても取得を行い、同様に“AIBO のプレゼンス”として他の一般ユーザに広告している。

5.3 実験評価

本節では、実装したアプリケーションを用いて、プレゼンス情報のコミュニケーションに与える影響を明らかにするためにを行った評価実験について述べる。

5.3.1 実験の目的と構成

本実験の主たる目的は、インフォーマルなコミュニケーションにおけるプレゼンス情報の与える影響を明らかにし、今後の検討を行うにあたっての判断材料とすることである。

今回の実験においては上記の目的を果たすために、ユーザが利用出来るプレゼンス情報の種類を変化させた場合に、コミュニケーションがどのように変化するかということを実験により明らかにする。実際に測定した項目は表 5.3 に示すとおりである。これらの項目を測定項目として選択した意図は以下の通りである。 C はコミュニケーションの量を最も直接的に反映すると考えられる。一方で、 N_i や L_i ($i = m, o$) は一回あたりのコミュニケーションがどの程度「盛り上がったか」ということの指標になると考えられる。また、 t が小さいことはユーザの行っているタスクの中でチャットの優先度が高いことを意味していると考えられ、そのような場合にはユーザの欲求に沿ったコミュニケーションが行われているということが期待出来る。

実験は表 5.4 に示す条件で 3 回、実験 1、実験 2、実験 3 の順に行った。それぞれの実験期間は平日の 3 日間であり、実験の被験者は全て本研究室に所属している学生 26 名である。被験者は全員互いに面識があり、約半数が柏の、残り半数が本郷の研究室に自分の席を持ち基本的にそこで研究を行っている。また、被験者のうち多くの学生が普段から MSN Messenger [19] 等のインスタントメッセージングソフトを日常的に用いており、インスタントメッセージによるコミュニケーションに慣れ親しんでいる。ただし、実験期間中に打ち合わせ等の理由により柏の学生が本郷へ、本郷の学生が柏へ来ることもたびたびあった。なお、学生の居室は柏と本郷にそれぞれ 2 室ずつあり、基本的に研究はこの居室で行われる。

それぞれの実験条件の設定は、以下のような意図による。すなわち、実験 1 では現行のインスタ

表 5.3: 測定項目

C	チャット回数
t	相手が最後に発言してから自分が発言するまでの時間差
N_m	1 回のチャットにおける自分の発言回数
L_m	1 回のチャットにおける自分の発言文字数
N_o	1 回のチャットにおける他人の発言回数
L_o	1 回のチャットにおける他人の発言文字数

表 5.4: 実験条件

	利用出来るプレゼンス情報
実験 1	スクリーンネーム及びステータスのみ
実験 2	なし (SIP アドレスのみ)
実験 3	本システムで実装した全ての情報

ントメッセージソフトと同様のプレゼンス情報を提供することで、これではなされるコミュニケーションは一般のインスタントメッセージソフトによるコミュニケーションとほぼ同等のものと考えられる。これに対し、実験 2 においては利用出来るプレゼンス情報を極限まで減らし、ユーザが相手ユーザについて知ることが出来る情報は、相手がオンラインかオフラインかと言うことだけである。これにより、プレゼンス情報が全く利用出来ない場合にコミュニケーションはどのような形になるのかということを知ることが目的である。一方、実験 3 では今回実装を行った全てのプレゼンス情報を利用し、結果を実験 1 や実験 2 と比較することで今回実装を行ったプレゼンス情報がコミュニケーションの支援にどの程度効果的であるかを観察する。

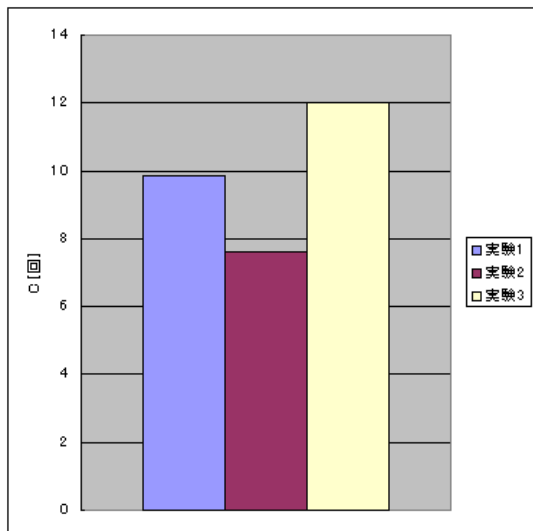
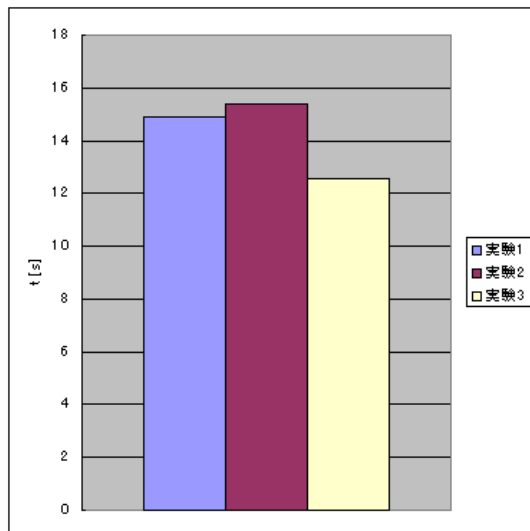
5.3.2 結果と考察

本節では、5.3.1 節で述べた実験の目的に対して得られた結果について述べると共に、実験結果について考察を加える。

実験の結果、得られたデータ集計したものを図 5.8 ~ 図 5.13 に示す。図 5.8 及び図 5.9 は、それぞれ C および t のユーザー一人あたりの平均値を示す。また、図 5.10、図 5.11、図 5.12 および図 5.13 はそれぞれインスタントメッセージによる会話セッション 1 回あたりの N_m 、 L_m 、 N_o 及び L_o の平均値を示す。なおいずれのデータも、それぞれの実験期間中に 1 回以下しかチャットによる会話をしなかったユーザは除外してある。

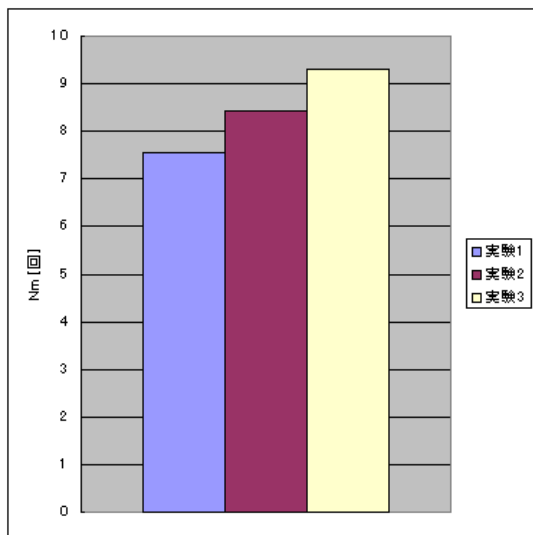
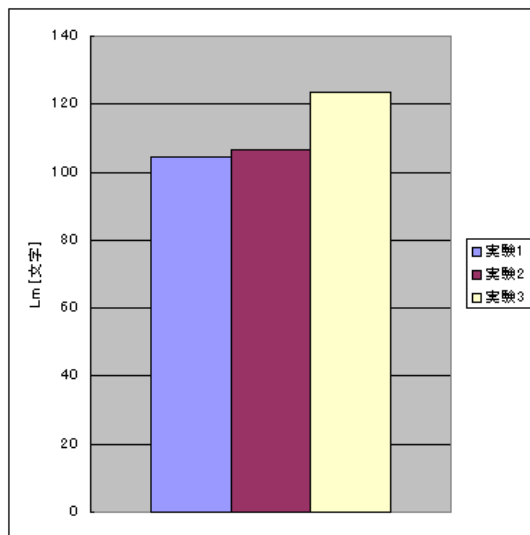
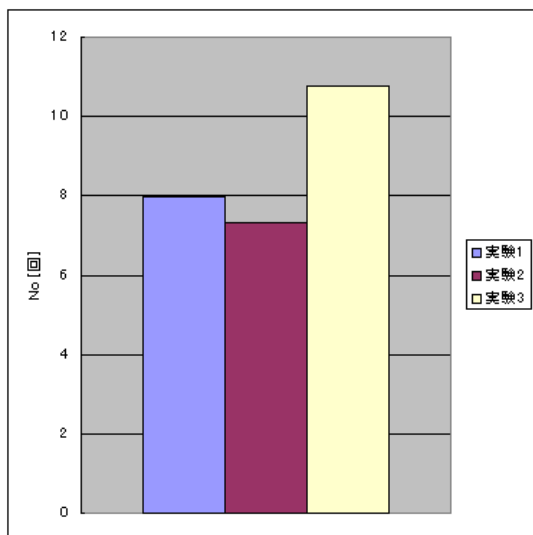
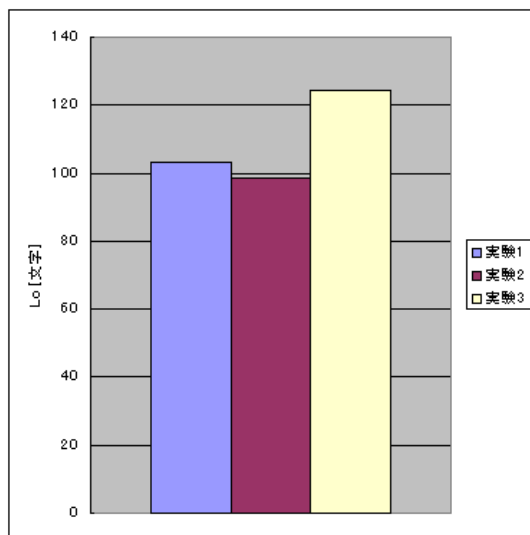
以下では得られたデータに対して考察を行う。

まず、それぞれの図より実験 2 において全体的にユーザ間のコミュニケーションが低調であったことがわかる。また、図 5.8 と図 5.10 ~ 5.13 を比較した場合、実験 1 は実験 2 や実験 3 と比べてチャットの回数自体は割合多いものの、チャット 1 回あたりの発言回数や発言文字数といったものはそれほど多くない。これは、実験 1 の場合は特に実験開始間もない時間帯において、被験者がシステムに慣熟していないためテスト目的でチャットを行ったり、あるいは被験者同士でアプリケーションの使い方について相談したりしていたことが原因として考えられる。このようなテスト目的や事務的連絡が目的で行われたチャットでは、そのチャットが行われている間に伝送された情報量 (バイト量) は比較的少量であることが想像され、用事が済んだ時点で速やかにコミュニケーションが終了しているものと思われる。その一方で、2 番目に行われてユーザがある程度アプリケーションの操作に慣

図 5.8: C の各ユーザあたりの平均値図 5.9: t の各ユーザあたりの平均値

熟していると思われる実験2では、 C は3つの実験中で最低となっているのに対して N_m と L_m は実験1よりも大きくなっている。これはコミュニケーションの回数自体は少ないもののそのコミュニケーションにおいて発言回数は実験1より多かったかということの意味している。実験2においては利用出来るプレゼンス情報は、相手ユーザがオンラインかオフラインかということだけであり、相手がどこにいるかという情報などは全く利用出来ない。このため、会話を行う際に通常のコミュニケーション内容に加えて、実験1では行われることの少なかった「今どこにいるのか」ということを始めとする相手の状況を問う発言が一定量付け加わったためではないかと考えられる。またこれは、実験を行った本研究室の居室が柏と本郷にそれぞれ2部屋ずつあり、特に柏の居室は2部屋が廊下で隔てられており相互に「向こうの部屋に今誰がいるか」ということを確認するのが困難であるため、例えば柏にいるユーザにとって、自分の部屋に姿が見えないがオンライン状態になっているユーザがいた場合、「自分がいない方の柏の部屋にいる」のか「本郷のどちらかの部屋にいる」のかが区別がつかないことも一因として考えられる。しかし以上の検討は推測に基づいており、さらなる正確な検討を行うためには会話内容を全て記録し、どのような会話が行われたかを分析する必要がある。その点については今後の課題としたい。

一方で、実験3においては実験1や実験2の場合と比較して、図5.8よりチャット回数が多いことが、図5.9より t の値が小さいことが見て取れる。これはコミュニケーションの回数が多いと共に、相手の発言を受け取ってから自分が発言するまで、あるいは逆に自分が発言を送ってから相手の返信が届くまでの時間差が少ないことを意味しており、コミュニケーションがより同期的に行われたと

図 5.10: N_m の 1 回あたりの平均値図 5.11: L_m の 1 回あたりの平均値図 5.12: N_o の 1 回あたりの平均値図 5.13: L_o の 1 回あたりの平均値

いうことを示している。また図 5.10～図 5.13 から、実験 3 の値は実験 1 や実験 2 よりも全体的に大きく、コミュニケーションが全体として活発だったことが推測出来る。このような結果となった要因としては、特に t の値の大小に関してはユーザが PC の前にいるかどうかというユーザのステータスが重要なことも勿論であるが、チャットで話すというコミュニケーションがユーザにとってどれほど重要性を持っているか、ユーザがコミュニケーションを何か他の作業を行いながら片手間に行うのではなくコミュニケーションを自身のメインの作業として行っているかと言うことに関連しているとも考えられる。そのような観点からは、実験 3 においてなされたコミュニケーションはユーザの興味に即したものであったということができ、BGM 情報や部屋の雰囲気といったプレゼンス情報によって、ユーザの興味に即した話題を提供することが出来たものと推測出来る。

5.3.3 得られたその他の知見

本節では、5.3.2 節で述べた項目以外に実験の結果得られた知見について述べる。

部屋の緊張度について、今回は各ユーザの申告値を各部屋毎に単純に加算平均を取る方法を用いたが、この方法では、

- 特に部屋の人数が多い場合に、一人が申告する“緊張度”を変化させても部屋全体の緊張度の変化量が少ない
- 申告する数値が数時間前に設定されたものでも数分前に設定されたものでも同様に扱われている

という点が問題点として実験の結果明らかになった。ゆえに、今後の課題として

- 加算平均に変わる何らかの平均値算出方法を検討すること
- それぞれのユーザが申告する数値を、最終更新時刻をキーとするなどして重み付けを行うこと

と言うことが挙げられる。

また、実験の被験者から実験終了後に寄せられた意見として、「キーボードやマウス操作が長時間行われていない場合に自動的にステータスを“退席中”にして欲しい」というものが多くあった。ユーザが PC の前を離れる場合には、自らの意志で能動的に席を離れる場合と誰かに呼ばれるなどして受動的に席を離れる場合の 2 通りに主に分けられる。能動的に席を離れる場合はともかく、特に受動的に席を離れる場合にステータスを“退席中”に変更するのを忘れて、あるいはいちいち操作するのが面倒くさいという声が多く聞かれた。このような機能は MSN Messenger [19] 等に実装されているが、今回は主に実装上の手間から見送ったものの、ユーザは「相手が退席中かどうか」ということから多くの情報を得ていることが明らかになり、ステータスを自動で変化させることの重要性が明らかとなった。

この他に、BGMを聴いているユーザの中からランダムに一人選び再生中の曲の情報をポップアップさせる機能について、「それを見てすぐに話しかけるということはあまり無かったが、友人達が普段どのような曲を聴いているのかがわかって興味深かった」という意見も寄せられた。チャットで直接話しかけることがあまり多くなかったという意味においては、このシステムを用いてのコミュニケーションの活性化には直接的には結びついていなかったが、長期的な視点においてはユーザ同士がより親しくなることを支援するという意味でインフォーマルなコミュニケーションの支援を行うことがある程度出来たと言えるのでは無からうか。

一方、今回の実験の問題点として、実験期間が短すぎたことが挙げられる。図 5.8～図 5.13 はそれぞれの値について平均値を示したものであるが、それぞれの値の分散はかなり大きく、標準偏差が平均値より大きくなった項目もあった。これは実験期間が、各実験においてそれぞれ3日ずつと短かったためと考えられる。チャットでのコミュニケーションにおいて、多くのユーザの関心を強く引く出来事・事件が主に実世界において生起しそのことに関する情報が断片的に他のユーザ（特に本郷の出来事の場合は柏にいるユーザ、逆も然り）に伝わった場合に、その話題についての会話では通常の会話の数倍～数十倍多くの発言がなされることがある。このような出来事・事件は毎日生起するわけではなく、「たまに」しか起こらないが、ひとたび生起してインスタントメッセージによる会話が開始された場合、その会話によるデータは t 以外の値を大きく引き上げる方向に作用する。今回の実験にこのような出来事がどの程度影響を与えたかは不明であるが、より実験期間を長くする事でこの種の出来事の影響は平滑化されるため、プレゼンスのみの影響をより正確に調べる事が可能になると思われる。

5.4 おわりに

本章では、まず5.2節で、プレゼンス情報のコミュニケーションに与える影響を評価すると共に、プレゼンス情報を用いてインフォーマルなコミュニケーションの支援を行うことの妥当性について確認するために実装を行ったアプリケーションについて報告を行った。次に、5.3節で実装したアプリケーションを用いて行った、プレゼンス情報のコミュニケーションに与える影響をを評価するための実験について述べるとともに、考察を行った。

第6章 結論

6.1 本論文の主たる成果

本研究では物理的に離れた空間に位置し、互いにフェイストゥーフェイスでのコミュニケーションを行うことが不可能であるユーザに対し、プレゼンス情報を活用することで日常生活の大部分を占めるインフォーマルなコミュニケーションの支援を目的としたシステムについて検討を加え、実装したテストベッドを用いてプレゼンス情報がどのようにコミュニケーションに影響を与えているかを評価する実験を行った。

フォーマルなコミュニケーションと異なり、インフォーマルなコミュニケーションは偶発的で話題も相手もコミュニケーションが実際に開始されるまでは不定であるという特徴があり、フォーマルなコミュニケーションには適している「必要なときに繋ぐ」というスタイルのシステムを用いることは不適當である。そこで本研究では「常につなげておく」というスタイルに注目し、これを実現するためにプレゼンス情報を有効に活用することで「繋がっている感」をユーザに与えることが効果的であると考えた。また、プレゼンス情報をインフォーマルなコミュニケーションの支援に用いる際に留意すべき点について検討を行い、それらの検討事項を反映させたシステムを実現するための手法について示した。

しかしプレゼンス情報というものは個人のプライバシーと密接に関連しており、むやみやたらと公開するわけにはいかない。またその種類も極めて多岐に渡るため、全てのプレゼンス情報を利用するという事は現実問題として不可能であり、どのようなプレゼンスを利用するかという取捨選択について検討を行うことが重要になる。この検討についての妥当性を確認するには実際にシステムを構築し、日常生活の中で実際に試用することが不可欠であるという考えに基づき、インフォーマルコミュニケーション支援システム ACDC の実装を行った。また、実装した ACDC システムを用いてコミュニケーションに対してプレゼンス情報がどのような影響を与えるのかという点について評価実験を行った。

今後のネットワーク環境のさらなる発展やコピキタスコンピューティング環境の実現により、地理的に分散した環境下のユーザ同士が日常的なインフォーマルコミュニケーションをネットワークを通じて行うことへの欲求はますます盛んになると筆者は考えている。そのような環境でユーザにとって有益なコミュニケーションを提供するには、プレゼンス情報を適切かつ効果的に用いる必要がある。そのようなときに、本研究で示したインフォーマルコミュニケーションの支援システムに関する検討及びプレゼンス情報のコミュニケーションに与える影響に関する考察が、ユーザに有益なコミュニケーションをもたらすことに貢献出来れば幸いであると考えます。

6.2 今後の課題

本研究では、ユーザのインフォーマルなコミュニケーションの支援に効果的であると考えられる、いくつかのプレゼンス情報をユーザに提供するシステムを提案した。それにより一定の効果をもたらすことには成功したが、フェイストゥーフェイスの環境と同等の環境を実現することにはまだ遠く及んでいない。今後はよりインフォーマルなコミュニケーションを支援するのに効果的であるプレゼンス情報の種類について、さらなる検討を行っていききたい。

また、実装を行ったアプリケーションを通じて実空間のユーザをネットワーク空間に投影することを目指しているが、現状ではそのインタフェースは AIBO を用いた以外は全て PC のモニタを用いている。しかし、ユーザが常に PC のモニタを見ているわけではないのは明らかであり、今後は PC 以外の情報出力装置を検討すると共に、ユーザの視覚以外の感覚に訴える Ambient Displays [17,31] 等との連携も模索していききたい。また、現状ではユーザのステータスをはじめ、ユーザがどの部屋にいるかなどの情報は多くがユーザの手入力に委ねられている。しかし、屋内におけるユーザ位置の検出システム [9,12,23] と連携することでこれを自動化することができ、よりユーザビリティの優れたシステムとすることが期待出来るため、こちらについても検討を行っていききたい。

謝辞

2年間研究を進めて行くに当たりまして、常日頃より有益かつ適切な御指導・御鞭撻を頂きました青山 友紀教授、森川 博之助教授に深く感謝いたします。また、研究を行う環境の整備に尽力していただいた助手の渡邊 廣次さん、王 溪さん、秘書の川北 敦子さん、宮島 史子さん、元秘書の松本 みどりさんに感謝いたします。

博士課程2年の三村 和さんには本研究のあらゆる局面において有意義な助言を頂きました。また、博士課程3年の川原 圭博さんには、ことあるごとに適切なアドバイスを頂きました。お二人には深く感謝いたします。また、私が修士1年の一年間、研究への取り組み方から内容に至るまであらゆることに対して御指導頂きました、本研究室OBの川田 雅人さんに深く感謝いたします。博士課程2年の金子 晋丈さん、岡 敏生さん、同1年の松本 延孝さん、川西 直さん、猿渡 俊介さん、修士課程1年の水野 浩太郎君、ジーン オリビエ カロン君、卒論生の神田 敦君には研究を進めていく上で様々な形でお世話になりました。深く感謝いたします。また、評価実験にご協力いただいた博士課程3年の橋口 知弘さん、同2年のゲン ホアイソンさん、林 素娟さん、修士課程1年の丸山 達也君、小森田 賢史君、平井 肇君、堀江 信吾君、木田 信雄君、オク ゼウク君、周 欣さん、卒論生の林 辰也君、中島 亮君、西浦 裕之君に深く感謝いたします。さらに、実験に協力していただただけではなく、同期として2年間苦楽を共にした飛岡 良明君、鯉江 尚央君、鹿島 拓也君、林 智天君、ヴー クァン ミン君に深く感謝します。良い同期に恵まれたおかげでこの2年間を乗り切ることが出来ました。この他にも、直接名を挙げることはいたしませんが多くの研究室のメンバーに様々な形でお世話になりました。心より感謝いたします。

また、ほとんど飲み会にしか顔を出さなかったにもかかわらず温かく迎えて頂き、殺伐としがちな研究生活に息抜きの時間をもたらせてくれたサークルをはじめとする友人達にも感謝いたします。

素晴らしいソフトウェアを作成し公開しているソフトウェア作者の方々、研究中に欠かせなかった素晴らしい音楽を創り出した数多くのアーティストの方々にも感謝します。どちらが欠けてもこの研究はなし遂げることは出来ませんでした。

みなさま、本当にどうもありがとうございました。

平成 17 年 1 月 31 日

参考文献

- [1] AIBO. <http://www.jp.aibo.com/>.
- [2] Ed. B. Campbell, J. Rosenberg, H. Schulzrinne, C. Huitema, and D. Gurle. *Session Initiation Protocol (SIP) Extension for Instant Messaging*, December 2002. RFC3428.
- [3] Sara A. Bly, Steve R. Harrison, and Susan Irwin. Media spaces: Bringing people together in a video, audio, and computing environment. *Commun. ACM*, Vol. 36, No. 1, pp. 28–46, 1993.
- [4] Mei Chuah. Reality Instant Messaging: Enhancing Intimacy Through Reality Streams. *Ubi-Comp2003*, October 2003.
- [5] P. Debaty and D. Caswell. Uniform Web Presence Architecture for People, Places, and Things. *IEEE Personal Communications*, August 2001.
- [6] Paul Dourish, Annette Adler, Victoria Bellotti, and Austin Henderson. Your Place or Mine? Learning from Long-Term Use of Audio-Video Communication. *Computer Supported Cooperative Work*, Vol. 5, No. 1, pp. 33–62, 1996.
- [7] Extensible markup language (XML). <http://www.w3c.org/XML/>.
- [8] Robert S. Fish, Robert E. Kraut, Robert W. Root, and Ronald E. Rice. Video as a technology for informal communication. *Commun. ACM*, Vol. 36, No. 1, pp. 48–61, 1993.
- [9] Yasuhiro Fukuju, Masateru Minami, Hiroyuki Morikawa, and Tomonori Aoyama. DOLPHIN: An Autonomous Indoor Positioning System in Ubiquitous Computing Environment. In *IEEE Workshop on Software Technologies for Future Embedded Systems(WSTFES2003)*, pp. 53–56, May 2003.
- [10] Rebecca E. Grinter and Leysia Palen. Instant Messaging in Teen Life. In *CSCW '02: Proceedings of the 2002 ACM conference on Computer Supported Cooperative Work*, pp. 21–30. ACM Press, 2002.
- [11] Mark Handel and James D. Herbsleb. What is chat doing in the workplace? In *CSCW '02: Proceedings of the 2002 ACM conference on Computer Supported Cooperative Work*, pp. 1–10. ACM Press, 2002.

- [12] Andy Harter, Andy Hopper, Pete Steggles, Andy Ward, and Paul Webster. The anatomy of a context-aware application. In *Mobile Computing and Networking*, pp. 59–68, August 1999.
- [13] Debby Hindus, Mark S. Ackerman, Scott Mainwaring, and Brian Starr. Thunderwire: a field study of an audio-only media space. In *CSCW '96: Proceedings of the 1996 ACM conference on Computer Supported Cooperative Work*, pp. 238–247. ACM Press, 1996.
- [14] ICQ Instant Messenger. <http://www.icq.com/>.
- [15] Ellen Isaacs, Alan Walendowski, and Dipti Ranganathan. Hubbub: A sound-enhanced mobile instant messenger that supports awareness and opportunistic interactions. *SIGCHI conference on Human factors in computing systems*, April 2002.
- [16] Ellen Isaacs, Alan Walendowski, Steve Whittaker, Diane J. Schiano, and Candace Kamm. The character, functions, and styles of instant messaging in the workplace. In *CSCW '02: Proceedings of the 2002 ACM conference on Computer Supported Cooperative Work*, pp. 11–20. ACM Press, 2002.
- [17] Hiroshi Ishii and Brygg Ullmer. Tangible bits: Towards seamless interfaces between people, bits and atoms. In *CHI*, pp. 234–241, 1997.
- [18] T. Kindberg, J. Barton, J. Morgan, G. Becker, D. Caswell, P. Debaty, G. Gopal, M. Frid, V. Krishnan, H. Morris, J. Schettino, and B. Serra. People, Places, Things: Web Presence for the Real World. *HPLabs Technical Report HPL-2000-16*, February 2000.
- [19] MSN messenger. <http://messenger.msn.com/>.
- [20] Bonnie A. Nardi, Steve Whittaker, and Erin Bradner. Interaction and Outeraction: Instant Messaging in Action. In *CSCW '00: Proceedings of the 2000 ACM conference on Computer Supported Cooperative Work*, pp. 79–88. ACM Press, 2000.
- [21] Elin Rønby Pedersen and Tomas Sokoler. Aroma: abstract representation of presence supporting mutual awareness. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 51–58. ACM Press, 1997.
- [22] Carman Neustaedter and Saul Greenberg. Balancing Privacy and Awareness in Home Media Spaces. *UbiComp2003*, October 2003.
- [23] Nissanka B. Priyantha, Allen K. L. Miu, Hari Balakrishnan, and Seth J. Teller. The cricket compass for context-aware mobile applications. In *Mobile Computing and Networking*, pp. 1–14, July 2001.
- [24] Anand Ranganathan, Roy H. Campbell, Arathi Ravi, and Anupama Mahajan. ConChat: A Context-Aware Chat Program. *IEEE Pervasive Computing*, July 2002.

- [25] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler. *SIP: Session Initiation Protocol*, June 2002. RFC3261.
- [26] Ylva H. Segerstad. Instant Messaging and Awareness of Presence in WebWho. *Proceedings of Mobilize! - 2nd Interdisciplinary International Workshop*, May 2001.
- [27] Skype. <http://www.skype.com/>.
- [28] The GNU oSIP library. <http://www.gnu.org/software/osip/>.
- [29] Mark Weiser. The computer for the twenty-first century. In *Scientific American*, pp. 94–104, September 1991.
- [30] Winamp com. <http://www.adcock8.freemove.co.uk/winamp.htm>.
- [31] Craig Wisneski, Hiroshi Ishii, Andrew Dahley, Matt Gorbet, Scott Brave, Brygg Ullmer, and Paul Yarin. Ambient Displays: Turning Architectural Space into an Interface between People and Digital Information. In *First Workshop on Cooperative Buildings (CoBuild '98)*, February 1998.
- [32] Melora Zaner, EK Chung, and Tammy Savage. 3rd and Net Generation : Designing for Inner Circles of Friends. *Intimate Ubiquitous Computing, UbiComp2003*, October 2003.
- [33] パンサエーンジューター, 川田雅人, 小野孝之, 森川博之, 青山友紀. コミュニケーションプレゼンスを考慮したブートストラップ機構. 電子情報通信学会ソサイエティ大会, September 2003.
- [34] 森川大補, 本庄勝, 山口明, 大橋正良. ユーザ状況に基づいた情報体系化とその利用に関する一検討. モバイルコンピューティングとユビキタス通信, September 2003.
- [35] 内田達人, 敷田幹文. 状況情報の利用による分散型コミュニケーション支援法の提案. 情報処理学会研究報告 2003-GN-49, October 2003.
- [36] 北陸先端科学技術大学院大学知識科学研究科. ナレッジ・サイエンス. 2002. <http://www.kousakusha.com/ks/index.html>.

発表文献

- [1] パンサェーン ジュター, 川田 雅人, 小野 孝之, 森川 博之, 青山 友紀. コミュニケーションプレゼンスを考慮したブートストラップ機構. 電子情報通信学会ソサイエティ大会, September 2003.
- [2] 小野 孝之, 川田 雅人, パンサェーン ジュター, 三村 和, 森川 博之, 青山 友紀. ユーザ主導型プレゼンスシステムの設計と実装. 電子情報通信学会総合大会, March 2004.
- [3] 小野 孝之, 三村 和, 川原 圭博, 森川 博之, 青山 友紀. インフォーマルコミュニケーションを支援するプレゼンス技術. 電子情報通信学会情報ネットワーク研究会, March 2005 (発表予定).
- [4] 小野 孝之, 三村 和, 川原 圭博, 森川 博之, 青山 友紀. プレゼンスを活用したインフォーマルコミュニケーション支援システムの試作. 電子情報通信学会総合大会, March 2005 (発表予定).