

修士論文 2003年度 (平成15年度)

知人関係を用いた  
プライバシー保護型マッチングシステムの研究

慶應義塾大学 政策・メディア研究科

氏名：須子 善彦

scottie@sfc.wide.ad.jp

## 修士論文要旨 2003 年度 (平成 15 年度)

### 知人関係を用いたプライバシー保護型マッチングシステムの研究

#### 論文要旨

本研究の目的は、ユーザ間の知人関係を活用することで、ユーザの情報開示に関するリスクの低減を実現する人材マッチングモデルを提案することである。

人材マッチングに限らず一般に、個人情報や機密情報などを開示することは、情報の不正利用をはじめとするリスクが生じる。一方で、そのリスクを恐れ情報を開示しなければ、多くの場合、必要とするサービスを十分に受けられない。また、開示しなくてはならない情報の種類や項目などはサービス提供者によって一方的に定義されるものが多く、情報開示者側の選択余地が限られてしまうという問題もある。

本研究では、(1) ユーザ間の知人関係という実社会上の資源をコミュニケーションツールにモデル化すること、及び、(2) そのモデル化においてユーザに対し情報開示における自主選択可能なオプションを提供すること、を通し、情報開示のリスクを低減するマッチングモデルを提案する。また、提案するモデルを、コストやスケーラビリティの点でユーザが広く利用可能なコミュニケーションツールとして実装し、実運用を通してモデルの有効性と課題を導き出す。

本モデルが想定するユーザは、業務の遂行など具体的な目標のために必要な知識や人材を検索するユーザである。また本モデルが特に有効に働くと想定される対象は、ユーザ間に高度な信頼関係が築かれ互惠性の高いコミュニティ、もしくは、構成員の間で共通の目的を共有し目的志向性が高いコミュニティである。

本モデルの特徴は、ユーザの情報開示に関するリスクの低減、ユーザ間で交換する情報の客観的な正確性の向上、及び、他の人材マッチングモデルと比較した実現可能性の高さである。

モデルの有効性と課題を導き出すため、人材マッチングを目的とする複数のコミュニティを対象に検証実験を行った。実験の結果、実装の安定性、モデルのスケーラビリティ、及び、有効性が検証された。また、人材の引き合わせ率(マッチ率)に関して、実用可能レベルに達する効果が示された。また、いくつかの課題が明らかとなった。

#### キーワード:

- 1: 人材マッチング    2: 知人関係    3: ネットワーク理論    4: グループウェア  
5: プライバシ    6: 自発性パラドクス    7: 信頼.

慶應義塾大学 政策・メディア研究科  
須子 善彦

## Abstract of Graduation Thesis Academic Year 2003

### The Research on the Privacy Secured Matching Model using Social Network.

#### Abstract

The purpose of this research is proposing the human resource matching model which reduces the risk about information disclosure of a user utilizing the acquaintance relation between users.

Not only in human resource matching but in any case, we generally take risks when disclosing personal information or confidential information. There is the risks of disclosed information being used by others arise. On the other hand, if one does not disclose their information, being afraid of the risk of it being used, they would not be able to receive the service they desire. Moreover, since the service provider decides which information should be disclosed, a person who receives the service information will be restricted to choose.

In this research, I propose the matching model which reduces the risks of the information disclosure through 1) modeling the acquaintance relation, which is the resources on the actual world, into the communication tool and 2) offering new options which are able to choose for users in information disclosure. Moreover, the proposal model is implemented as a communication tool which users can use generally in respect of cost and scalability and the validity and the subject of the model are drawn thorough practical use.

The users whom this model assumes are people with certain goals and objectives, looking for information or person to solve their task. The range of community of users that this tool is assumed to be especially effective, would be a community of high reciprocity nature which is with a confidential relation advanced among users, or a community with the high purpose intention nature and where a common objective is shared by the participants.

The features of this model are reduction of the risk about information disclosure of a user, the improvement in the objective accuracy of the information exchanged among users, and the high possibility of it being carried out in real life compared to other matching models.

In order to evaluate the validity and to conduct the subject of a model, the verification experiments were carried out in several communities aiming at human resource matching. The stability of this implementation, the scalability of this model, and validity were verified as a result of the experiment. Furthermore, the effect of reaching an usable level was shown about matching rate. Besides, some issues that need to be improved in the future were found.

#### **Keywords:**

1: Human Resource Matching    2: Social Network    3: Network Theory  
4: Groupware    5: Privacy    6: Paradox of Voluntary    7: Trust

**Yoshihiko SUKO**  
**Graduate School of Media and Governance**  
**Keio University**

# 目次

<b>第1章 序論</b>	<b>1</b>
1.1 背景	1
1.1.1 目的に応じた自由なランデブーの実現	1
1.1.2 ネットワーク社会とリソースマッチング	2
1.1.3 人材マッチングへの要求事項	2
1.2 目的	5
1.2.1 知人ネットワークを用いたマッチングモデルの実現	5
1.2.2 知人ネットワークが生み出す信頼	6
1.2.3 知人関係と取引コスト	7
1.2.4 紹介	8
1.2.5 マッチングモデルの研究における本研究の位置づけ	8
1.2.6 本研究の意義	9
1.3 用語等の定義	9
1.4 本論文の構成	11
<b>第2章 関連研究</b>	<b>12</b>
2.1 関連研究の概要	12
2.2 協調作業や知識共有に関する関連研究	12
2.2.1 協調作業や知識共有を行うためのコミュニケーションツール	12
2.2.2 コミュニケーションツールの性質の変化	14
2.3 人材マッチングに関するコミュニケーションツール・サービスの事例	15
2.3.1 ツールやサービスの事例	15
2.3.2 事例における課題点	16
2.4 知人ネットワークを用いたコミュニケーションツール	18
2.4.1 知人ネットワークを用いたコミュニケーションツール	18
2.4.2 知人ネットワークを用いた人材マッチング	19
2.4.3 事例における特徴と課題点	21

<b>第 3 章</b>	<b>知人関係を用いたマッチングモデル Degrees Connect</b>	<b>23</b>
3.1	前提条件	23
3.2	モデルの詳細	23
3.2.1	知人ネットワークの定義	23
3.2.2	知人ネットワークの管理	24
3.2.3	知人リンクの構築プロセス	24
3.2.4	知人リンクの編集	25
3.2.5	知人ネットワークを用いた情報配信	26
3.3	モデルの特徴	30
3.3.1	情報開示リスクの管理	30
3.3.2	情報の信頼性の向上	31
3.3.3	高い実用性と広い応用範囲	32
<b>第 4 章</b>	<b>Degrees Connect の設計</b>	<b>35</b>
4.1	システムの全体構成	35
4.2	ユーザの個人情報の取得機能	37
4.3	ユーザの知人リンクの取得機能	37
4.3.1	ユーザの知人リンク取得に関する機能群	37
4.3.2	知人関係の認識機能	38
4.3.3	知人関係の承認機能	40
4.3.4	知人リンクの構築機能	41
4.3.5	知人リンクの編集機能	42
4.4	知人リンクの蓄積機能	43
4.5	知人ネットワーク処理機能	43
4.6	知人ネットワークを用いたメッセージ処理機能	44
<b>第 5 章</b>	<b>Degrees Connect の実装</b>	<b>45</b>
5.1	実証ターゲットと実装ポリシー	45
5.2	実装環境	45
5.3	知人ネットワークの構成要素の実装	47
5.4	ユーザの個人情報管理に関する機能	50
5.4.1	ユーザの個人情報の処理	50
5.4.2	ユーザの個人情報の取得	52
5.5	ユーザの知人リンクの取得機能	53
5.5.1	ユーザの知人リンクの管理機能群	53
5.5.2	ユーザの知人リンク取得の認識機能	54

5.5.3	知人リンクの構築機能	57
5.6	知人ネットワークのネットワーク演算	57
5.7	メッセージの送受信	61
5.7.1	人材検索ユーザからの人材検索メールの送信	61
5.7.2	受信ユーザに提供される機能	62
5.8	人材検索メールのメッセージフォーマット	65
5.9	アプリケーション層	66
5.9.1	アプリケーション層の実装	66
5.9.2	Web 版実装	67
5.9.3	携帯電話版実装	68
<b>第 6 章</b>	<b>Degrees Connect の評価</b>	<b>72</b>
6.1	要求定義の確認	72
6.2	モデルの妥当性に関する先行実験	73
6.2.1	実施要領	73
6.2.2	調査結果	73
6.2.3	先行研究の結論	74
6.3	実証実験の目的と実施概要	74
6.3.1	実証実験 1(ISAGA2003) の実施概要	74
6.3.2	実証実験 1(ISAGA2003) における調査方法	75
6.3.3	実証実験 1(ISAGA2003) の実施結果	76
6.3.4	実証実験 2(HCD) の実施概要	76
6.3.5	実証実験 3(SIV Networking Seminar) の実施概要	77
6.3.6	実証実験 3(SIV Networking Seminar) における調査方法	77
6.3.7	実証実験 3(SIV Networking Seminar) の実施結果	78
6.3.8	実証実験 4(SIV Business Networking) の実施概要	78
6.3.9	実証実験 4(SIV Business Networking) の調査方法	79
6.4	実験結果と検証項目の考察	80
6.4.1	実装の動作検証	80
6.4.2	モデル評価 1; 人材検索メールの配信範囲の妥当性に関する検証	81
6.4.3	モデル評価 2; 情報開示における自発性パラドクスの解消	85
6.4.4	マッチ率への効果の定量評価	87
6.4.5	マッチ率への効果の検証結果と考察	88
6.4.6	その他; モデルの利点、問題点の把握	90
6.4.7	ケース; 仲介者の声	92

第7章 結論	94
7.1 本研究の達成事項	94
7.1.1 人材マッチングにおける自発性パラドクスを解消	94
7.1.2 エンドユーザ主体の情報管理の実現	94
7.1.3 共通の知人（仲介者）が生み出す信頼性の高い人材マッチング	94
7.1.4 人材マッチングの効果について	95
7.1.5 高い実現性（スケーラビリティ、コストパフォーマンス）	95
7.2 未達成及び不確定事項と今後の課題	95
7.2.1 モデル化における単純化と有効性の範囲について	95
7.2.2 実社会の事象のモデル化の精度について	96
7.2.3 自主選択可能な選択肢に関して	97
7.2.4 実装の改善	97
7.3 今後に向けて	98
謝辞	99

# 目 次

1.1	情報開示とマッチングの精度	4
2.1	エキサイトフレンズ	16
2.2	従来のモデル	17
2.3	Microsoft 3 °(Three Degrees)	19
2.4	Acquaintance Network System	20
2.5	Stanford Alumni InCircle	21
3.1	信頼判断の委譲モデル	24
3.2	知人リンクの構築プロセス	25
3.3	知人リンクの削除と禁止	26
3.4	要件の送信	28
3.5	要件への返信	29
3.6	共通の知人の識別情報	29
3.7	ランデブーの確立後	30
3.8	情報開示リスクの管理	31
3.9	情報の信頼性の向上	32
4.1	システムの全体構成	36
4.2	知人リンクの構築プロセス	43
5.1	実装の全体像とレイヤリング	47
5.2	ユーザの個人情報クラス群の関係	51
5.3	ユーザ情報管理ユーティリティクラス群の関係	52
5.4	ネットワーク管理ユーティリティクラス群の関係	53
5.5	ネットワーク演算処理クラス群の関係	58
5.6	Web 版実装のページ構成	69
5.7	Web 版実装のインターフェース	70
5.8	携帯電話版実装のページ構成	71
5.9	携帯電話版実装のインターフェース	71



6.1	クレーク数による到達可能なノード数 . . . . .	82
6.2	全ユーザに対する到達可能なユーザ数の累計比 . . . . .	83
6.3	ホームカミングディ2003の知人ネットワーク . . . . .	84
6.4	ユーザ毎の知人リンク数 . . . . .	84

# 表 目 次

4.1 ユーザから取得する個人情報項目 . . . . .	37
4.2 知人リンク取得に関する機能群 . . . . .	38
4.3 知人リンク管理インターフェース . . . . .	40
4.4 知人リンク構築要求レコード . . . . .	41
4.5 知人リンクの構成要素 . . . . .	42
4.6 知人ネットワーク処理に関する主要な機能 . . . . .	44
4.7 知人ネットワークを用いたメッセージ送信機能 . . . . .	44
5.1 実装の種類と利用するターゲット . . . . .	45
5.2 実装環境 . . . . .	46
5.3 core.framework.network.Node インターフェース . . . . .	48
5.4 core.framework.network.Link インターフェース . . . . .	49
5.5 ユーザの個人情報の分類 . . . . .	50
5.6 core.framework.foundation.DCUserCert インターフェース . . . . .	51
5.7 core.framework.foundation.UserUtilsFactory 抽象クラス . . . . .	52
5.8 core.framework.foundation.UserUtils インターフェースの一部 . . . . .	53
5.9 core.framework.network.NetworkUtilsFactory 抽象クラス . . . . .	54
5.10 core.framework.network.NetworkUtils インターフェースの主要なメソッド . . . . .	55
5.11 知人リンク構築要求テーブル friend_auth のスキーマ . . . . .	56
5.12 知人リンクテーブル friends のスキーマ . . . . .	57
5.13 core.framework.engine.DCMatchingEngine インターフェース . . . . .	58
5.14 core.framework.engine.MatchingEngineFactory インターフェース . . . . .	59
5.15 core.logic.network.DConnectFOAFLinkPipe クラス . . . . .	60
5.16 core.framework.mail.FPMailLogic インターフェース . . . . .	62
5.17 受信ユーザに提供される機能 . . . . .	63
5.18 返信識別パラメータの書式 . . . . .	64
5.19 人材検索メール保持テーブル connect_msg_hashes のスキーマ . . . . .	65
5.20 人材検索メールのメッセージフォーマット . . . . .	66

6.1	検証実験における検証項目 . . . . .	72
6.2	先行実験の実施要領 . . . . .	73
6.3	実証実験 1(ISAGA) の実施要領 . . . . .	74
6.4	実証実験 2(HCD) の実施要領 . . . . .	76
6.5	実証実験 3(SIV Networking Seminar) の実施概要 . . . . .	78
6.6	実証実験 4(SIV Business Networking) の実施概要 . . . . .	79
6.7	実証実験 1 のデータ . . . . .	88
6.8	実証実験 4 のデータ . . . . .	89

# 第1章 序論

## 1.1 背景

### 1.1.1 目的に応じた自由なランデブーの実現

#### 目的達成とランデブー

インターネットの発展は、地理的制限を越えたグローバルなコミュニケーションを可能とした。TCP/IP ネットワークに参加する全てのノードは、通信相手の識別子（IP アドレス）さえわかれば、地球上の至る場所から、いかなる距離を越えてコミュニケーションを開始できる。しかし、このグローバルな到達性の実現は、我々人間がコミュニケーションに求める本質的な要求の一部に過ぎない [1]。

我々人間は社会的存在として、あらゆる活動においてコミュニケーションという行為を行っている。逆に言えば、社会的存在としての人間のあらゆる活動の目的達成のためには、コミュニケーションという手段が必要である。その手段 = コミュニケーションを実現するためのインフラストラクチャの設計において重要視すべき点は、その手段に対応した目的である「人間の社会的活動における目的達成」を支援するものでなくてはならない点である。我々人間の日常活動における目的達成を実現する手段として、グローバルな到達性の上に、更なるコミュニケーションインフラストラクチャの改善が必要である。

その改善の一つとして、ランデブーの自由度の向上が挙げられる。ユーザの目的達成にとって、いかに適切なコミュニケーション相手（ノード、ユーザ、情報のソース）の存在を把握し、その存在に対してアクセスを行い、コミュニケーションを確立するか、といった課題は常に存在する。本論文では本課題をランデブー確立の課題という。世界中の誰とでも（少なくとも先進国の間では）コミュニケーションを取ることが技術的に可能になった今、次に掲げる問題として、誰とコミュニケーションを取るべきか、に課題の焦点が当てられてきた。コミュニケーション相手の把握、選定、アクセスといったランデブーの確立を行う方法に関して、ユーザに与えられた選択肢の充実化とユーザによる自由な選択が必要である。

インターネットは、その設計思想からユーザ間の通信や情報伝達を高度に制御するスーパーバイザーは存在しない [2]。ランデブーの確立は、エンドユーザが自分自身の責任で行う必要がある。インターネットのようなオープンでフラットなコミュニティでは、エンド

ユーザが自己責任にてコミュニケーション相手を選定するために必要な判断材料は少ない。実際にはユーザは判断材料として、インターネット上の資源だけでなく、自己の経験やコミュニケーション相手の社会的地位・肩書き、人間関係といった実社会上の資源を用いてコミュニケーション相手を選定している。

したがって、いかに実社会上の資源をインターネットにおけるコミュニケーションモデルに取り込むか、という点はランデブーの自由度の向上にとって重要な視点である。

### 1.1.2 ネットワーク社会とリソースマッチング

前世紀から続いてきた交通や流通に関する技術の進歩と、近年の情報通信技術の急速な発展により、今日は、モノ・人・情報が、世界規模で行き来する時代となった。このようにスピード化・流動化が進んだ社会においては、社会の構成要素の多くが、様々な他の要素と複雑にコミュニケーションを取りながら、社会を構成している [3][4]。

例えば近年、従来の縦型組織に代わってプロジェクト型組織というものが注目されている [5]。このプロジェクト組織とは、解決しなくてはならない課題が発生した時点で、その課題解決のために必要な人材を集めて一時的な組織を作り、目的達成と同時に解散するという性質を持つ組織で、日々刻々と変化する課題に対応するために、メンバーシップにオープン性・流動性を持たせていることが特徴である。

したがって、プロジェクト型組織では、チームビルディングの段階で、その成果が大きく左右されると言われる。目的達成のために、必要なスキル、知識を持った人材をいかに集めてチームを構成するかが重要となってくる。

同様の例として、学際研究が挙げられる。従来の学問領域では扱いきれない問題を、複数の学問領域、専門領域にまたがる見地、知識を持った人材によって解決するのが学際研究である。この場合も、解決する問題に対して、いかに必要な人材を集めるかが重要である。

他には、フリーエージェントへの注目 [6] や、近年のオープンソースコミュニティ [7] 等も同様の例として挙げられる。

そこで、目的達成のために必要な人や情報といったリソースを、より信頼性の高い形で、リスク無く獲得したい、といったニーズ、つまり、人材マッチングに対するニーズの高まりは、今日の社会的要請の一つである。

### 1.1.3 人材マッチングへの要求事項

#### ユーザの情報開示量とマッチングの精度

人材マッチングでは一般に、マッチングする人材間で、お互いに関する情報が詳細に交換され、お互いの理解度が高くなるほど、より良いマッチングが行われる可能性が高くな

る。ここでいうより良いマッチングとは、マッチングの目的に対して適材な人材が見つかることである。

例えば、デザイナーを必要としている人が単に「デザイナー」という単語のみで必要とする人材を表現していた場合、本当はキャラクターデザインが必要であるのに、それが不得意な Web デザイナーが引き合わされてしまうといったことがある。この場合、人材を必要としている人がより自身のニーズを詳細に表現していれば、或いは、デザイナー側も得意とする分野についての情報を詳細に告知していれば、ミスマッチが防げていた可能性が高い。

他には、より詳細な病歴を開示すれば、よりよい医師が診察してくれる、或いは、よりよく調剤された薬を貰えるといったことも考えられる。

これらの例に限らず一般に、ユーザの個人情報の開示度と、それによって得ることの出来るマッチングの精度は、正の相関関係にある。さらに一般化すると、エンドユーザが開示する自身の個人情報と、受けることが出来るサービスは、正の相関関係にある。

### 情報開示度とリスク

前項で指摘したように、情報開示度を大きくすればそれだけ精度の高いマッチングを実現できる。しかしエンドユーザは、「必要なサービスに対して開示する情報の度合いは可能な限り最低限にしたい」という要求を持つ。情報開示にはリスクが伴うためである。

先ほど述べた病歴といった情報は、個人によっては、公開することによって精神的な苦痛を感じる、或いは、その後の日常生活において不都合が生じるといったことが考えられる。これらセンシティブ情報を扱う際は、開示する対象を制限したいといったニーズがある。

別の例を挙げる。あるプロジェクトにおいて技術者が不足し、その技術者を探す必要が生じたとする。このとき、このプロジェクト技術者を募集しているという事実が、多くの人に知れ渡るほど、一般により多くの技術者が見つかる可能性が高くなる。Web ページにプロジェクトの詳細と募集する技術者の要件を載せることで、理論的には世界中の人がその事実アクセスすることができる。

一方で、多くの「余計な人」にプロジェクトの詳細や、そのプロジェクトが技術者を必要としている事実（すなわち、そのプロジェクトは技術者不足で困っているという事実）が知られてしまう。ここで言う「余計な人」とは、この場合におけるプロジェクトにとって必要な存在（すなわち、技術者あるいは技術者を紹介してくれそうな第 3 者等）以外の人である。この余計な人に知られてしまうことは、不都合が発生するかもしれない。たとえば、プロジェクトの詳細に、知的財産や機密事項に関わる可能性がある場合や、ライバルのプロジェクトに技術者不足で困っているという事実が伝わってしまうといった不都合である。

つまり、自主的に情報を開示することによって、その情報を得た他者が一方的に得をし、自分自身が不利になってしまう、といった現象が起こる。また、開示した情報を不正に利用されるといったリスクも発生する。情報開示に関して、フリーライダーが発生し、いわゆる「自発性パラドクス」[8]といわれる現象、つまり情報に関する「共有地の悲劇」[9]が起こりうる。特にインターネットにおける情報開示は、個人特定が困難であるため、「自発性パラドクス」が起こりやすいと考えられる。

#### 人材マッチングへの要求事項

1.1.3 で述べた情報開示におけるリスク = 「自発性パラドクス」の発生を恐れ、人材を必要としている事実を伝える範囲を狭くしてしまえば、1.1.3 で述べた関係によって、適材な人材が見つかる可能性は低くなる。

したがって、人材マッチングには相反する 2 つの要求、すなわち「人材を検索したい者の情報開示量を下げながら、目的達成に従ってよりの確な相手とのマッチングを可能にすること」が存在する。

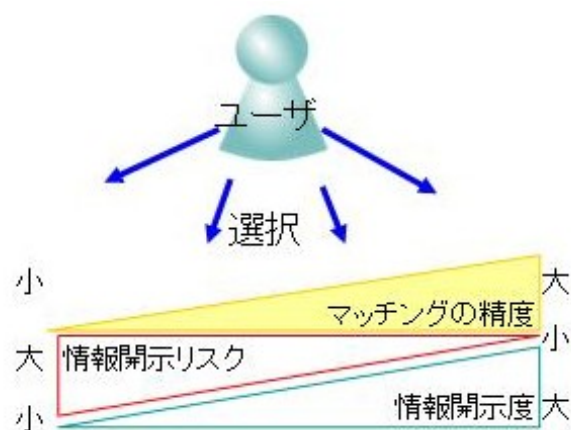


図 1.1: 情報開示とマッチングの精度

大半がサービス提供者側で決められている

多くのサービスは、そのサービスを受けるために、ユーザが開示しなくてはならない個人情報の開示度を規定するプライバシーポリシーが存在する。それらポリシーの多くは、サービス提供者によって策定され、ユーザに一方的に提示されることがほとんどである。そして、エンドユーザはサービスを受けるためには、そのプライバシーポリシーを受け入れざるを得ない。なんらかの理由によりユーザがそのプライバシーポリシーを受け入れることができなければ、多くの場合、サービスは受けられない。したがって、必要なサービスレベルを保ちながら、必要最低限のプライバシー開示度を実現するユーザの自主選択可能な選択肢は、一般的に少ない。例えば、就職支援サイトは一種の人材マッチングを提供するサービスであるが、大半の就職支援サイト [10][11] は、ユーザ登録時に数ページに渡る膨大な個人情報を登録しなければ利用が出来ない。

つまり、どの情報を開示するかは、サービス提供者側によって一方的に決められているのが今日の一般的な現状であり、また提供後の情報の扱いもサービス提供者側の管理に委ねるしかない。ユーザがそれを望まなければ、情報の提供を行わないという選択肢が存在するが、その場合サービスをまったく受けられないことがほとんどである。

したがって現状は、サービス提供者側の要求に従い個人情報を提供しサービスを受ける選択肢か、サービスを受けられない代わりに個人情報を開示しないという選択肢の、正反対の 2 つの選択肢しか存在しない。

よって、ユーザが持つニーズである、「必要なサービスを受けつつも、個人情報の開示量を最低限にする」ユーザが自主選択可能な選択肢は少ない。

## 1.2 目的

### 1.2.1 知人ネットワークを用いたマッチングモデルの実現

本研究の目的は、知人ネットワークにおける信頼の結節点の存在を用いることで、情報における自発性パラドクスを解消するマッチングモデルを実現することである。

1.1 では、スピード化、複雑化する社会の中で、我々の社会的活動の目的達成のために、人および人が持っている情報へのアクセス、それらとのマッチングが重要であると述べた。また、ユーザの情報開示に対するニーズとマッチングの精度との間にはトレードオフが存在し、そのトレードオフにおいて、ユーザが自主選択可能な選択肢が少ないことを指摘した。また、情報開示には自発性パラドクスというリスクが存在することで、マッチングには、相反する 2 つの要求を満たす必要があることを述べた。

そこで本研究では、実社会上における資源であるユーザ間の知人関係をコミュニケーションツールにモデル化すること、及び、そのモデル化においてユーザに対し情報開示におけ



る自主選択可能なオプションを提供することを通し、情報における自発性パラドクスを解消するマッチングモデルを提案する。また、モデルをシステムとして実装し、実運用を通して有効性と課題を明らかにする。

## 1.2.2 知人ネットワークが生み出す信頼

1.1.1 にて述べたようにインターネットには、スーパーバイザーは存在しない。プライバシーや機密情報の管理に関して、統一された仕組みがあるわけでもなければ、損害補償を行ってくれる保険機関もない。すべてはユーザの自己責任である。また、コミュニケーションの内容や主体に対して、意図的な詐称があっても、それをチェックし修正させる強制権をもった主体は存在しない。すべてのコミュニケーションは自己責任である [12]。

人材マッチングにおいてもその点は同様である。人材マッチングにおいて検索された相手が、自分の目的達成において、相応しい人材かどうか、求められる役割期待に応える人材か、問題を起こしたりしないか、といった点は、自分自身で判断する必要がある。

したがって、従来では閉じられたメンバーシップの中、つまり長い経験によって担保された信頼性の中から、適材を見つけ出してくることが多かった。しかし、先に述べたように、今日は、より多様な人材が必要であり、従来の手法では限界が出てきている。

一方で、インターネットのようなオープンでフラットなコミュニティにおいて、信頼性を判断する材料は少ない。

信頼性の担保、つまりこの場合では、人材マッチングの目的達成にとって、検索された相手が相応しいかどうか、といった、コミュニケーション相手の Identify と検証において現在広く取られている手法の一つとして、信頼できる権威者或いは権威のある機関にお墨付きを貰うという手法がある。例えば、パスポートは国家という権威機関が所持者の Identity にお墨付きを与えている。同様のモデルを用いたものが、PKI[13] である。

もう一方の手法として、不特定多数の第三者による評判を用いた信頼性の担保が挙げられる。見る映画を選ぶとき、洋服を選ぶときなど、第三者による評判がその選択に影響することは、我々の日常生活において経験することである。

この2つの手法の中間に位置する手法として、自分自身が信頼する特定の知人の評判が挙げられる。この手法は、先に述べた手法と異なり、信頼の担保を行う権威者が広く一般に知れ渡った存在ではなく、自分自身にとっての「権威者」である点である。本来自分自身が判断すべき信頼判断の根拠を、自分が信頼できる他者に委譲するモデルといえる。

## PGP が提案する信頼のネットワーク

先に述べた中間的手法を取り入れたものとして、PGP[14] の公開鍵に対する署名モデルがある。このモデルは、公開鍵暗号方式における公開鍵の信頼性（本当に本人のものであ

るか)を保証するため、その信頼性を担保してもらうための署名を他者から貰う、といった仕組みである。

宮川 [12] では、この仕組みを拡張し、マルチホップで信頼性判断のための基準を提供する PGN というものを提案している。

これらのモデルが用いている、人々の関係をグラフとして捉え、人々のやり取りや信頼性といった関係の強さを計算するという考え方は、グラフ理論の世界で行われてきた。また最近では、グラフ理論に、グラフの成長といった概念を取り入れたネットワーク理論が盛んであり、先に述べたスピード化、複雑化する今日において、注目すべき考え方の一つとなっている<sup>1</sup>。

### 1.2.3 知人関係と取引コスト

人材マッチングには、情報と情報とのマッチングとはじめとする他の一般的なリソースマッチングと比較して特徴的な性質がある。例えば、Web ページを検索する際、検索エンジン [16][17] などを用いるが、その結果は検索エンジンのアルゴリズムに基づいて、検索に用いた要件 ( 現行の検索エンジンにおいては主に、検索に用いられる検索キーワード ) に近い順にリストアップされる。したがって一般的にユーザは、検索結果リストの上にあるページの方が自分の検索要求に見合ったページであると判断する。また、翻訳機能を提供する Web サービスを検索する場合は、その翻訳 Web サービスが持つ辞書の登録語数や一般的な利用における誤訳率などを見て、少しでも登録語数の多いもの、かつ、誤訳率が少ないものを選択するのが一般的である。

しかし、人材検索の場合は事情が少々異なる。例えば、翻訳を頼みたいとする。これが Web サービスであれば先に述べたことと同様に、翻訳に関する性能が少しでも高いほうがより望ましい。同じ考えを人間の場合にも当てはめたとすると、TOEFL[18] や TOEIC[18] といった英語検定のスコアが少しでも高い人間ほうが望ましい。899 点と 900 点であれば、後者を選ぶ。しかし、検索相手が Web サービスではなく人間である場合は、以下のようなことが起こりうる。890 点の知人と、900 点の見知らぬ他人では、前者を選ぶことが起こりうるのである。支払わなくてはならない報酬や、その他、翻訳業務を委託する契約においてなんら変わりが無いとしてもである。

このような判断が行われるのは、Web サービス等機械のパフォーマンスと比較して、人間のパフォーマンスは、不確実性が高いためである。人間を選ぶ場合は、その人が正しく要求を満たしてくれるか、手抜きをしないかどうか、不当な理由で納期を延ばさないかどうか、業務委託において得た情報を他者に漏洩しないかどうか、といった様々な不確実性を考慮してパフォーマンスを予測しなければならない。したがって、英語の能力検定のス

<sup>1</sup> グラノベッターが [15] にてとらえた人間関係はグラフ理論的なものであったといえる

コアが多少低くとも、知人を選ぶ方ことで、過去の経験や、今後も続く知人関係といった要素などから、不確実性を下げることが出来るならば、そちらを選択するのである。知人を選択する方が、取引コスト [19] が低くなると判断することが多い。

#### 1.2.4 紹介

ある物事を達成する上で人材が必要となった場合、まず自分の知人の中に探している人物がいないかどうか探ることがある。そして、知人の中に求める人材がいない場合、次のような疑問を持つことがある。「自分が探している人材が、自分の「知人の知人」の中に、もしかしたら存在するのではないか」といった疑問である。

知人から探すというのは、前項で述べた理由による利点が大きいためであると考えられるが、知人の中に求める人材が見つからない時に、上記の疑問を持つということは、次のことが考えられる。自分の「知人の知人」という対象に対しても、自分が知人に対して抱く取引コスト上の利点と同様の効果を求めていると考えられる。このことは、立場を変えて考えてみると分る。自分の知人 A が、人材を探しているとする。求める人材はなかなか見つからない。そのような中、自分の他の知人の中に、知人 A が求める人材に近い人 B が存在したとする。そのような際、自分は、その両者を互いに紹介することで、引き合わせることが出来る。両者が引き合わされることで、自分、或いは、どちらか片方に不利益が起るような場合でなければ、自分がその両者を互いに紹介することは、自分と両者の 3 者にとって利益である。また、紹介するのは、自分の知人同士なので、前節において述べた不確実性に関して、自分は自信と責任を持って両者に保障できる。紹介される側からみれば、不確実性が低くなるのである。

このことは、前項において述べた不確実性の解消に対して、知人関係が有効に働いていることを証明する身近な現象である。

#### 1.2.5 マッチングモデルの研究における本研究の位置づけ

人材マッチングモデルの研究は様々なところで行われている。中には社会システムを大きく変更することで実現するモデルも少なくない [20][21]。その様な中、私はインターネット技術とネットワークコミュニケーションを研究する立場から、個人が容易に利用することが可能なインターネットのコミュニケーションツールとして、前項で述べたユーザの情報開示リスクを低減する人材マッチングモデルを実現する。

### 1.2.6 本研究の意義

本研究は、実社会上における資源であるユーザ間の知人関係をコミュニケーションツールにモデル化すること、及び、そのモデル化においてユーザに対し情報開示における自主選択可能なオプションを提供することを通し、情報における自発性パラドクスを解消するマッチングモデルを実現する。

知人関係が生み出す「情報開示に関するリスクの低減」といったメリットを実現し、1.2.4 で述べた紹介という事象にまつわるユーザのニーズを満足させる。

本研究で提案するマッチングモデルが用いる「知人関係という実社会上の資源」によって、情報開示の管理や信頼性の判断を行うことは、前節までに述べてきたように、我々の日常において頻繁に行われている事象である。したがって、本研究で提案するマッチングモデルは、多くの人々にとって経験的に理解しやすいモデルであると考えられる。また、本モデルは、人材間のマッチングに限らず、あらゆる種類のリソースマッチングにおいて活用できる。具体的には、プライバシー保護と信頼性が高く要求されるシチュエーション、例えば、健康や福祉、教育や人材などの分野での人材マッチングやリソースマッチングに対して広く応用できると予想される。また、モデルを自律分散型に単純化した結果、スケーラビリティやコストの面での利点を利点を実現した。また、社会システムを大きく変化させる必要が無い場合、多くの人材マッチングモデルと比較して、実用可能性が高い。

## 1.3 用語等の定義

本節では、本研究で特別に定義が必要な用語について説明する。

知人リンク：

本モデルにおいて、相互の信頼関係を持つユーザ間の関係

知人ネットワーク (Social Network)：

ユーザを節 (ノード) とし、前述の知人リンクを辺 (リンク) とするネットワーク。

実社会上の：

本研究が提案するマッチングモデル、並びに、その実装システムの外部のすべての世界を指す言葉として本研究では便宜的に用いる

プライバシー :

基本的人権の核になる部分であり、個人の私的領域

個人情報 :

プライバシーのうち、社会的関係性において個人から外部に公開された、もしくは社会的行動によって外部に放出されたことにより、外部から認識されるもの

識別情報 :

個人を個人と特定し、他者から識別するための個人情報。具体的には、氏名、メールアドレス等

センシティブな情報 :

個人におけるプライバシーや、組織における知的財産、機密情報といった、不特定多数に開示されることで、精神的な苦痛を感じる、或いは、その後の日常生活において不都合が生じたり、経済的な損失を受ける情報

プライバシー保護型 :

本研究の目的である情報開示に関する自発性パラドクスの解消によって開示リスクが低減される情報を、本研究では便宜的に広義のプライバシーと定義する。具体的には前述のセンシティブな情報を指し、これらの情報の開示リスクを低減するマッチングを、プライバシー保護型マッチングと便宜的に定義する

ランデブー :

コミュニケーションにおいて、コミュニケーション相手の存在を把握、選定し、その存在に対してアクセスを行い、1対1のコミュニケーションを行うこと

コンタクト :

人材検索メールに興味を持った人が送信ユーザに対して返信し連絡を取ること

マッチ :

本モデルにおける人材検索にて、2 ユーザ間でニーズが一致し、協調作業を開始する段階に移行すること

## 1.4 本論文の構成

本論文では、第 2 章において本研究の先行研究ならびに関連研究、関連事例について述べる。

第 3 章においては、本章で定義した目的を達成するために、本研究で提案するマッチングモデルについて説明する。

第 4 章では、具体的なユースケースから要求仕様を定義し、要求仕様に沿って第 3 章で述べたモデルを実現化するための設計について述べる、第 5 章にてモデルの評価のために行ったプロトタイプ実装について説明する。

その後、第 6 章にて、プロトタイプ実装の運用を通して、モデルの評価を行う。第 7 章にて、評価の考察と今後の課題について述べた後に、本論文をまとめる。

## 第2章 関連研究

### 2.1 関連研究の概要

本章では、本研究の関連研究として、人材マッチングや人材を中心としたリソースマッチングを行うためのコミュニケーションツールの関連研究やサービスについて述べる。

これらのツールは、目的別に分けると、直接人材マッチングに特化しているものと、ユーザ間やグループにおける協調作業や知識共有を目的としているものに分けられる。また、本研究においても注目している知人ネットワークを用いたコミュニケーションモデルは、協調作業や知識共有を目的としている分野で数年前から研究されてきているが、最近では人材マッチングに特化した Web サイトなどにおいても、知人ネットワークを用いる関連研究が登場してきた。

### 2.2 協調作業や知識共有に関する関連研究

#### 2.2.1 協調作業や知識共有を行うためのコミュニケーションツール

協調作業や知識共有を行うためのコミュニケーションツールは、一般的にグループウェアやコラボレーションウェアと呼ばれる。

グループウェアとは、名前の通りある特定のグループ内において、協調作業を支援する、或いは、知識の共有を促進するためのコミュニケーションツールである<sup>1</sup>。一般に、コミュニケーション機能としては、Web メール、電子会議室、掲示板などを備えており、スケジュール、To-Do リスト、アドレス帳などをグループで共有する機能を持つ。また、会議室予約、購買予約など、組織マネジメントを支援する機能を搭載することが多い。

コミュニティウェアの定義は様々であるが、本論文では、グループウェアと比較して、よりオープンなメンバーシップを対象とし、コミュニティに見られる構成要因間のダイナミズムやインタラクションによる創発・自己組織化といった現象を創造、強化し、コミュニケーションそのものを発生させる方向にコミュニティを誘導・支援するソフトウェアを指すものとする。

---

<sup>1</sup>グループウェアという考え方は、1960年代にアメリカのスタンフォード研究所の Douglas C. Engelbart が提唱したとされ、コンピュータを使って人間の知的能力を増幅させるという考えを元にした遠隔共同作業支援システム「NLS (ONLine System)」の実証実験が行われている。

この分野のツールやサービスは非常に多く存在するため代表的なもの、或いは、本研究と関連性の深いものだけを紹介する。

#### 専用サーバ/クライアント型グループウェア

社内 LAN などに専用のサーバを設置し、独自のソフトウェアをクライアント毎にインストールする形のグループウェアで、グループウェアの黎明期の大半を占めた。代表的なものとしては、Lotus Notes[22] や Microsoft Exchange[23] がある。

これらは、高機能で設定が複雑であり、高価であったため容易に導入できるものではなく、大企業といった大きな組織において主に使われた。したがって、主にメンバーシップが固定的で管理されているコミュニティにおいて、協調作業支援や知識共有の促進を目的に使われることが多かった。

#### Web グループウェア

インターネットの発展によって、専用のソフトウェアやプロトコルを用いる 2 層クライアントサーバ型システムの多くは、Web ブラウザと HTTP を用いる 3 層クライアントサーバモデルに移行した。グループウェアにおいても同様に、サイボウズ Office[24] や iOffice[25] といった Web ベースのグループウェアが登場した。これらは、導入作業や初期設定が容易であり、比較的安価であったため、小さなグループにおいても導入できるものであった。これらの登場により、グループウェアは、中小企業や NPO、研究コミュニティ、プロジェクト組織といった一時的なコミュニティ等における協調作業支援や知識共有の促進を目的としても使われることが多くなった。

#### ASP グループウェア、コミュニティウェア

Web グループウェアの普及とほぼ同様の時期に、広く個人ユーザに対してグループウェア的機能を提供する ASP 型のサービスが登場した。例えば、eGroups[26] はグループ内でメーリングリストやファイル共有、スケジュール共有といったグループウェア的機能を利用できるサービスを提供する ASP である。ユーザは Web ブラウザさえあれば、少ない知識で無料にて利用できるため、協調作業の支援や知識共有の促進といったグループウェアの恩恵を、同僚や友人同士といった小さなコミュニティにまで広げることが可能にした。国立情報学研究所の新井紀子助教授と(株)NTT データポケットが手がける NetCommons[27] は、教員グループや市民団体等といったインターネット技術やセキュリティの知識がないユーザコミュニティを対象に、安全かつ簡単に情報共有することを支援するためのシステムである。



また、知識の共有といった分野では、About[28] や OKWeb[29]、ひとびと net[30]、K スクエア [31]。といった Q&A をベースとした ASP サービスが登場した。これらは、ある問題への解決にあたって、必要な情報や知識を持っている人を探し出し、その人から情報や知識を伝えてもらうものである。

## P2P グループウェア

ASP 型グループウェア、コミュニティウェアの次に登場したのが、peer to peer グループウェアである。

peer to peer 型のコミュニケーションツールの特徴は、導入に対してサーバを必要としないことや、ユーザの増加に対するシステム運用のコスト増加が小さいといった特徴がある。一方で、クライアントサーバモデルと比較して一般的にネットワーク資源を多く消費する。したがって、エンドユーザが利用できるネットワークが広帯域化した近年において、急速に注目されてきた。代表的なものとしては、Groove Workspace[32] や ArielAirOne[33] といったものが挙げられる。

先ほど述べた特徴から、Web グループウェアと同様、主に中小企業や NPO、研究コミュニティ、プロジェクト組織といった一時的なコミュニティなどにおいても協調作業支援や知識共有の促進を目的として使われ始めている。

また、peer to peer 型のコミュニケーションツールとして、MSN Messenger[34] や AOL Messenger[35] といったインスタントメッセンジャが挙げられる。実際の通信の形としては Pure 型 peer to peer となっているものは少ないが、バディリストを用いたメッセージの交換や、ゲーム、アプリケーションの共有といったコミュニケーションモデルのユーザへの見せ方は、peer to peer 型コミュニケーションツールといえる。また、groove にはインスタントメッセンジャにおけるバディリストのようなリストでグループを管理する。グループウェアにインスタントメッセンジャのコミュニケーションモデルが導入された事例として注目できる。また、主要なインスタントメッセンジャの一つである MSN Messenger の最近の機能強化を考えると、両者の未来は同じものになる可能性は低くない。個人単位の協調作業を容易に実現するツールが今後普及していくと予測される。

### 2.2.2 コミュニケーションツールの性質の変化

第 1 章で述べたように、社会におけるコミュニティの役割の変化に従い、先に述べたコミュニケーションツール・サービスの役割にも変化が見られてきた。

従来、人間は目的達成のために、組織に入る、或いは、組織を構成することで、組織のリソースを利用して目的達成を行うことが多かった。したがって、組織活動を支援することが、人間は目的達成の支援の主な方法論だった。しかし近年、社会のスピード化、個人が

扱える情報量の増加、及び情報処理技術の進化が進み、個人の能力が拡張された結果、目的達成に対して数人の個人が目的に応じたグループをダイナミックに形成し、目的達成を行うというスタイルが増えてきた。従来に比べて、グループ形成の過程が目的達成に対して果たす役割が大きくなった。したがって、コミュニケーションツール・サービスも、グループ形成の過程を支援するものが増えてきた。コミュニティウェアが目指すところにも、同様の変化が見られる。

次節からは、それら個人間のマッチングを中心とする人材マッチングツール・サービスを紹介する。

## 2.3 人材マッチングに関するコミュニケーションツール・サービスの事例

### 2.3.1 ツールやサービスの事例

人材マッチングの分野におけるコミュニケーションツール・サービスの先駆的存在として、米国のコミュニティサイト Bolt[36] が挙げられる。このサービスは、登録ユーザ毎に、Web 専用メールボックスやスケジュール機能、掲示板機能などが組み合わされたプライベートページを発行する。ユーザはユーザ登録時に、趣味や性別、年齢など自分に関する情報を登録する。この情報は、bolt を使っている他のユーザに公開される。この公開された情報（プロフィール）に対して、ユーザは相互に検索しあい、お気に入りの人に対してアプローチしてゆく。この際、多数対多数のコミュニケーションモデルである掲示板やグループチャットと、1対1のモデルであるインスタントメッセンジャなど、アプローチのための様々な機能を提供している。ニックネームによるコミュニケーションや Web 専用のメールボックスの提供など、ユーザのサービス外での識別可能性を下げるために、現在のマッチングサービスにおいて広く利用されているこれらの仕組みは、bolt においても実現されていた。

国内のサービスではエキサイトフレンズ [37] が有名である。このサイトも bolt の影響を受け、基本的な機能は同じであり、携帯電話からの利用が可能である。また、これらの汎用的な人材マッチングに加え、ジャンル別に特化したサービスも登場してきている。Sports Match Onlin[38] は、ユーザ登録の際、スポーツ名、スポーツ歴などを登録することで、パートナーを検索できるサービスである。

また、人間そのもののマッチングではなく、人間が持っている情報に注目し、効率の良い情報共有のためにユーザ間の知人関係を活用しているマッチングシステムも存在する。

情報提供者の能力を管理する仲介サーバを設け、情報消費者の要求にオンデマンドに引き合わせる方式をとっているのが Matchmaker[39] である。Yenta[40] では、メールや個人が管

理する文書などから抽出した文章の意味を処理し、興味の近いユーザのクラスタを自動生成することで、ユーザが相互にコンタクトを取れるようにするシステムである。CoMeMo[41]では、コミュニティ内の暗黙知を形式化させるシステムで潜在的な共通点を探す。



図 2.1: エキサイトフレンズ

### 2.3.2 事例における課題点

これらのコミュニケーションツールやサービスにおいて共通する問題は、コミュニケーションモデルが一点集中型で、かつ、「ミクロなコミュニケーション」(後述)が上手く取り入れられていない点である。そして、これらのモデルは、ツールの設計者、サービス提供者が設計したものであり、ユーザはそれに対して従わざるを得ない。

これらのコミュニケーションモデルでは、一端ユーザから発信された個人情報は、一般に同じサービスを利用する他のユーザから自由にアクセスできる。したがって、ユーザが個人情報を開示することによって生じる、個人情報の漏洩や不正な再利用といったリスクは大きい。また、このモデルにおいては、情報を発信するユーザとその情報を享受するユーザの間に、情報開示リスクの不平等が生じている。情報開示のリスクは、本来、人材マッチングの結果引き合わされる 2 者間でできるだけ平等にリスクが配分されるべきである。しかしこれらのモデルにおいては、既に発信された他のユーザの個人情報を、ユーザが検

索を行っても、ユーザに対して個人情報の開示リスクの追加コストは生じず、先に個人情報を発信したユーザの開示リスクだけが大きくなる。

また、個人情報は記録されると、システム内に蓄積・保持されるため、他者の情報を検索することの利益と、自己の情報を更新することの間に強い関係性がないため、情報が常に更新されないことが多い。

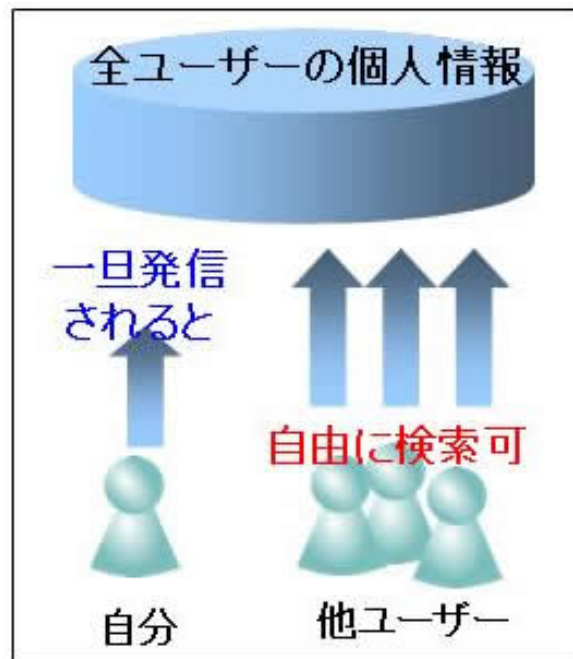


図 2.2: 従来のモデル

第1章で述べたように、人材マッチングにおいては、相反する2つの要求、すなわち「人材を検索したい者の情報開示量を下げながら、目的達成に従ってよりの確な相手とのマッチングを可能にすること」が存在する。これに対して、先ほど述べたモデルでは、ユーザの個人情報は広くユーザに公開されるため、情報開示量は大きい。

また、目的達成に従った的確な相手とのマッチングに関しても、第1章において、人間のパフォーマンスは、その不確実性の高さから、より近い関係の深い人間を的確な相手として選ぶ可能性が高いことを述べた。このような、より近い関係の深い人間とのコミュニケーション＝「ミクロなコミュニケーション」は、これらのモデルでは考慮されていない。

## 2.4 知人ネットワークを用いたコミュニケーションツール

### 2.4.1 知人ネットワークを用いたコミュニケーションツール

本研究では、前節までの課題の解決手法として、知人ネットワークにおける信頼の結節点の存在を用い情報における自発性パラドクスを解消するマッチングモデルを実現する。そこで本節では、同様にユーザ間の知人関係を活用したコミュニケーションツールについて紹介する。

ユーザ間の知人関係を活用する一番身近なコミュニケーションツールは先ほども触れたインスタントメッセンジャや peer to peer グループウェアである。これらは、ユーザの情報の開示範囲やコミュニケーション対象を、パディリストという知人関係を用いて管理する。

インスタントメッセンジャは、当初はユーザ間の即時的な同期型コミュニケーションを実現するものであったが、近年はユーザ間でアプリケーションを共有する、或いは、ファイルを共有する、ビデオチャットをする等、グループウェア的な方向に機能が強化されている。Microsoft は、自社のインスタントメッセンジャMSN Messenger と、Peer to Peer 実装の Advanced Networking Pack for Windows XP[42] と協調して動作するソフトウェア 3 °(Three Degrees)[43] によって、知人関係を活用した新しいグループウェア的機能を実現している。このソフトウェアを利用するユーザ同士は、3 °グループという名称の知人グループを作成すると、同一グループの中で、所有している音楽コレクションを共有することができる。但し、Napster[44], Gnutella[45] をはじめとする従来の Peer to Peer ソフトウェアによる音楽共有とは、共有を実現する概念が異なる。Microsoft 3 °(Three Degrees) は、友人という限られたメンバーシップの中で、一つの音楽を共有するが、このときの共有は、友人同士がそれぞれの音楽を持ち寄って同じ部屋に集まり、一つのプレイヤーを共有しているという状態である。したがって、誰か一人でも、曲変更や早送り、巻き戻しといったプレイヤー操作を行うと、他の友人へも影響する。また、後で聴くために、友人の音楽コレクションに入っている曲を保存しておく、といったことは出来ない。

Microsoft 3 °(Three Degrees) は、友人グループという概念を用いることで、従来では「共有 = 公衆送信」であった音楽共有を、ユーザの「好きな音楽を共有したい」といったニーズを大きく制限することなく実現した。音楽の配信範囲の制限を、友人グループというユーザの知人関係を用いて実現している。

PGP も知人関係を用いたコミュニケーションツールといえる。PGP は公開鍵暗号を実現するフレームワークおよびその実装である。PGP では公開鍵暗号に用いる公開鍵の所有者認証において、ユーザ間の署名を用いて実現している。公開鍵暗号方式を用いたコミュニケーションにおいて、コミュニケーション相手の公開鍵が本当の自分が実際にコミュニケーションを取りたい相手の公開鍵であるかどうかを確認できなければ、成りすましによる情報漏えい等のリスクが生じる。したがって、公開鍵の正当性（本当にその人の公開鍵

であるかどうか)を保証する仕組みが必要となる。この仕組みにおいて、PKIX と並ぶ仕組みが、PGP が採用した方法である。具体的には、ある公開鍵が自分のものであるということを証明する際に、知人に署名をもらうことで、その保証してもらうというものである。知人による署名によって、コミュニケーション相手の特定に関する信頼性を向上している。



図 2.3: Microsoft 3.0 (Three Degrees)

#### 2.4.2 知人ネットワークを用いた人材マッチング

先ほど述べた PGP の仕組みを、人材マッチングに応用した研究として、PGN[12] が挙げられる。PGP の署名の仕組みは自分と知人との関係だけを扱うが、PGN は、この関係を「知人の知人」、「知人の知人の知人」という具合に、2 クリーク以上の範囲に拡張し人材マッチングに応用したものである。

また、ANS(Acquaintance Network System) と呼ぶ情報共有の機構 [46] では、知人の関係を、意味と重みを持ったパイプと呼ばれるデータ構造で表現し複数のパイプの合成処理により新しい関係に対応するパイプを生成する。この機構により、人が知人のつてをたどって情報提供者にたどり着くのと同様に、限定した範囲に開示された個人情報を用いて、広範囲の情報提供者と情報消費者の引き合わせを実現する。

このシステムでは、各個人に対応したパーソナルマネージャというエージェントが存在し、電子メール等、ユーザ間のコミュニケーション媒体の内容を形態素解析エンジンによ

て分析することで、パイプの生成と、リクエストに対するサービスを行っている。また、IKNOW[47]は、さらに知人の自己登録などを行うことで精度を高めている。また、コラボレーションの視覚的な検索、確認、分析ができ、ナレッジネットワークの成長を時間軸でたどれる。

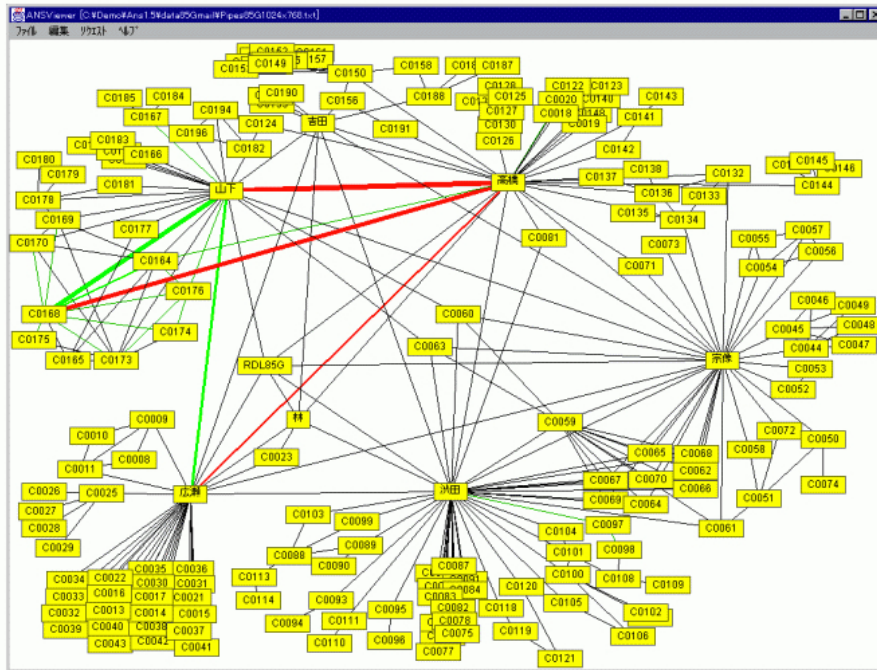


図 2.4: Acquaintance Network System

また、ユーザ間の共通の関心によって、ユーザ間を引き合わせるというユニークなコンセプトのサービスとして、関心空間 [48] が挙げられる。また、日記中の言葉をキーワード化することで日記の内容が意味的に近い日記同士を引き合わせるはてなダイアリー [49] の機能も似たようなコンセプトのサービスである。

2003 年に入り米国では、プロフィールマッチング（日本でいう出会い系サイトのようなもの）のサービスとして、ユーザ間の知人関係を利用することでユーザの趣味・嗜好や顔写真といった情報を交換する範囲を設定することができるサービスが試験的に行われている [50][51]。ユーザは、「友人の友人までには、自分のプロフィールを公開する」といった設定ができる。従来の大半のプロフィールマッチングでは、他のユーザに対して検索対象となる情報は、全てのユーザに開示されており、全てのユーザから条件によって絞り込むというモデルが一般的であったのに対し、知人関係を活用したことで実社会上の文脈にそって情報検索の範囲を設定することを可能にしている。

また、米スタンフォード大学のオンライン Alumni サービスの一つに InCircle[52] という

ものがある。スタンフォード大学の Orkut 氏が開発するマッチングエンジンを用いたサービスであり、スタンフォード大学の Alumni 向けに提供されている人材検索サービスである。このサービスは、登録者が自身の交友関係をユーザ間同士で公開・共有することで、Alumni 同士の交流を盛んにするサービスである。このサービスの提供をきっかけに、Alumni の OB 会運営に対する寄付といったコミットメントの増加が見られたという。

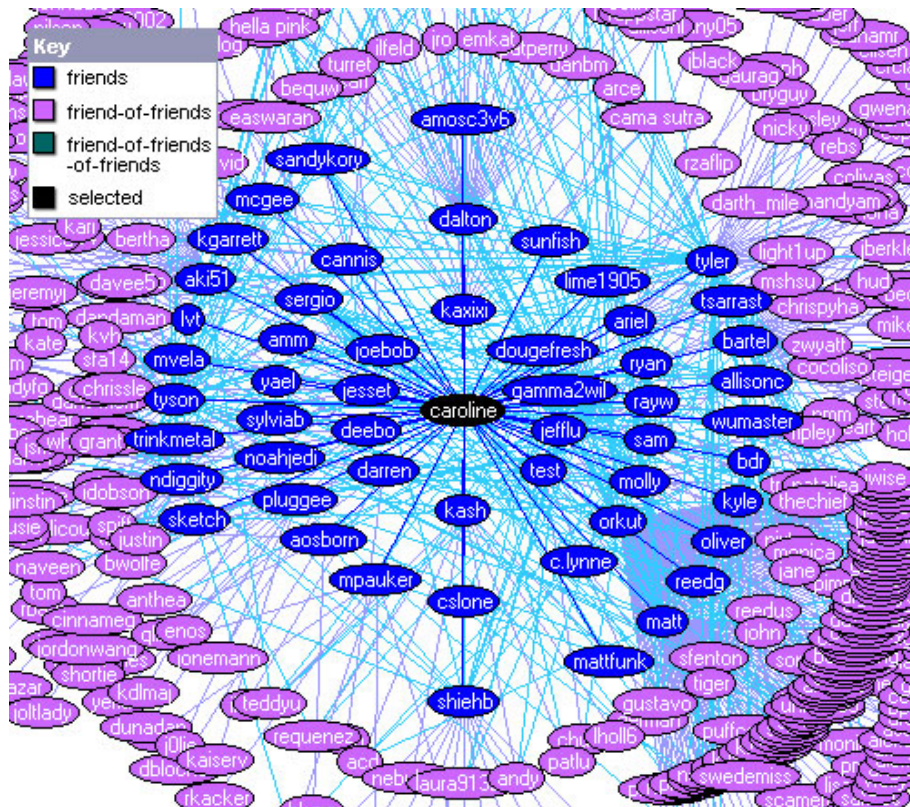


図 2.5: Stanford Alumni InCircle

これらの関連研究は、知人関係といった実社会の資源を用いることで、情報検索の手段を新たに提供している点が共通点である。一方で、本研究との差異は、本研究が主にプライバシーや機密情報の開示リスクの低減といったユーザのコミュニケーションを行う際の安全性のために、ランデブー精度の向上をうたっている点である。

### 2.4.3 事例における特徴と課題点

これまで述べてきたように、ユーザ情報の開示範囲の制限、情報検索の範囲の制限、或いは、コミュニケーション相手の選定、といった目的等に、ユーザ間の知人関係を利用するという試みが、2003 年度に入り、様々な分野で行われはじめてきた。

このような中、本研究は、ユーザが情報開示するリスクの低減のために、自己情報コン



トロール権の強化に注目し、モデルを確立する。また、モデルの単純化によって、コストパフォーマンス、スケーラビリティといった実用可能性の高いモデルを提案する。

先ほど紹介した事例では、モデル自身の複雑さや、モデルを実現するために必要な事前準備が大きい。したがって、ユーザが気軽に利用をすることが困難である。また、ユーザがやり取りするコミュニケーション内容がシステムに伝わってしまう、或いは、ユーザが構築する知人ネットワークの情報が他のユーザに開示されてしまう。

また、これらのマッチングモデルは、2.3.1 にて紹介したものも含め基本的に、ユーザの保有スキルや趣味・嗜好に関する情報、所属組織や最終学歴といったユーザ自身の属性を、ユーザに登録させる。そして、人材検索においては、引き合わされる 2 者間のニーズの一致を、登録された属性の情報に従って判断するのが一般的である。

この方法は、人材検索を自動化することもでき、その点においてはユーザの負担を減らすものであるが、最初の時点で属性を登録する手間や登録した情報の漏洩リスク等のコストが発生する。また、登録する属性の項目はサービス提供者によって決められてしまうため、ユーザの自由度が制限される。サービス提供者の想定外のユースケースに対応できない可能性がある。さらに、登録された属性によって、ニーズの一致に関する判断を行うため、その方法（アルゴリズム等）にユーザが関与することが出来ず、ミスマッチが発生する可能性がある。

## 第3章 知人関係を用いたマッチングモデル

### Degrees Connect

#### 3.1 前提条件

本モデルは、実社会上における資源であるユーザ間の知人関係をコミュニケーションツールにモデル化すること、及び、そのモデル化においてユーザに対し情報開示における自主選択可能なオプションを提供することを通し、情報における自発性パラドクスを解消するマッチングモデルである。

ユーザ間の知人関係を取得し、その関係にそって情報を配信する。このことにより、ユーザ情報の開示範囲を最適化しながらユーザ間のランデブーを確立、ユーザの情報開示リスクを低減する。

本モデルはコミュニケーションモデルの特性から、ユーザ間に高度な信頼関係が築かれ、互恵性の高いコミュニティ、もしくは、構成員の間で共通の目的を共有し、目的志向性が高いコミュニティで特に有効に働く。

例えば、共通の組織への所属したメンバーで構成される同窓会や、学会など同じ関心領域で人が集まっている Community of Interest といわれるタイプのコミュニティなどである。

#### 3.2 モデルの詳細

##### 3.2.1 知人ネットワークの定義

本モデルでいう知人ネットワークとは、ノードを本モデルの利用ユーザ、リンクをノード間の知人関係とするネットワークである。

本モデルでリンクとして定義される知人関係は、利用するユーザによる主観的な判断によって形成される知人関係と定義する。つまり、ユーザ自身の主観的判断による知人関係によって作られる人脈のネットワークが知人ネットワークとなる。

本モデルは、第1章において述べたように、情報配信の範囲の制限およびコミュニケーション相手の選定における信頼判断をユーザの知人に委譲するモデルである。したがって、ユーザは、信頼判断を委譲しうる信頼関係を持つ人物を知人として登録することが要求されている。基本的には、この登録によって形成される知人関係を、本モデルでは知人リン

クと定義し、知人ネットワークにおけるリンクと定義する。ただし、この登録には双方向の承認が必要である。登録時に、相手のユーザから承諾されないと、本モデルでいう知人リンクの形成はされない。つまり、双方向の信頼関係が確認されなければ、本モデルにおける知人リンクとは定義されない。

自分が信頼する人に、信頼判断を委譲するモデル;  
 「自分が信頼する人が信頼する人」  
 → 自分もある程度信頼できるという考え



図 3.1: 信頼判断の委譲モデル

### 3.2.2 知人ネットワークの管理

前項で述べたようにユーザは、信頼判断を委譲しうる信頼関係を持つ人物をあらかじめ登録する必要がある。人材マッチングを応用対象とする本研究においては、その信頼判断を委譲しうる信頼関係とは、ユーザ自身が自分の責任で「相互に紹介可能な信頼できる友人」を登録することとする。

ユーザは、自身の知人を登録するバディリストというリストをユーザ毎に個別に持つ。ユーザは自身のバディリストを常に参照することが出来、操作を行うことが出来る。主な操作は、新たな知人の登録、既に登録された知人の解除である。

### 3.2.3 知人リンクの構築プロセス

ユーザは新たな知人を登録する際、登録したい知人に関する識別子を指定して、登録を行う。具体的な識別子としては、E-Mail アドレスやローカルのネットワークシステムにおいて識別可能なアカウント名である。この識別子を用いて、ユーザは登録する知人に対して、登録する旨を通知する。登録要求を受けた知人は、その要求に承諾を行うか、拒否をする。

承諾する際は、登録要求を行ったユーザ宛に承諾通知を行う。承諾をした時点で、双方向の信頼関係が存在するということになり、知人リンクが形成される。

拒否する際は、登録要求通知に対して、明示的に拒否の行為を行うか、何の操作も行わない。何の操作も行わないことで、知人リンクが形成されず、事実上の拒否を意味する。

なお、ユーザは登録要求を行った時点で、自身のバディリストに、登録承諾待ち状態として登録される。登録要求を送った相手が明示的に拒否をしない限り、相手の名前は登録承諾待ち状態のままリストに登録される。

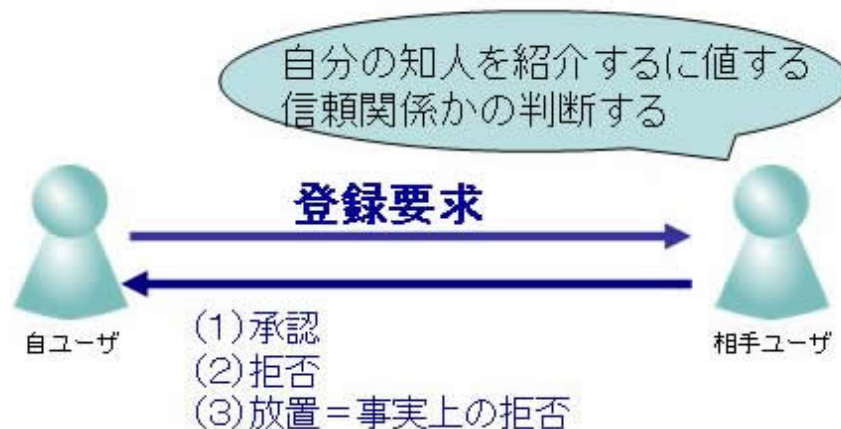


図 3.2: 知人リンクの構築プロセス

#### 3.2.4 知人リンクの編集

ユーザは任意の時に、自身のバディリストを編集できる。具体的には、さらに新しい知人を登録する操作、及び、一度登録した知人を削除する操作である。

新しい知人を登録する操作は前項にて説明した操作と同様である。一度登録した知人を削除する操作は、バディリストから行う。削除したい知人を選択し、削除操作を行うと、知人リンクの定義である「双方向の信頼関係」が解除され、知人ネットワーク上のリンクが削除される。ただし、この際、ユーザ間の実社会における人間関係への影響が起きる可能性があるため、関係の解除が相手にも通知される「削除」という操作に加え、削除された相手のバディリスト上は見た目の変化が起きない「禁止」化という操作も、削除操作のひとつのオプションである。

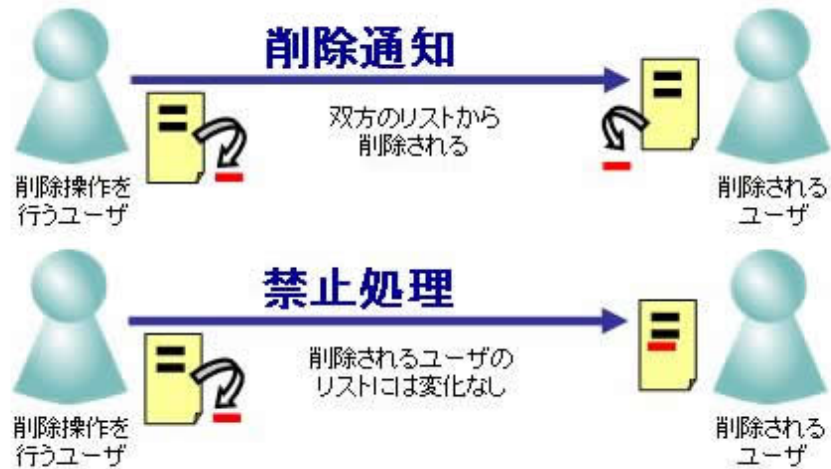


図 3.3: 知人リンクの削除と禁止

### 3.2.5 知人ネットワークを用いた情報配信

#### (1) 要件情報の作成

最初に、人材を検索したいユーザー（以下、検索ユーザー）は、目標達成のために必要な知識や人材に関する情報を、要件情報として作成する。例えば、プロジェクトチームの結成の際などにある分野の専門技能や専門知識を持った人材が必要である、ある困難な課題の解決のために必要な知識や情報をもった人物を知っている人を探したい、急用でいけなくなったコンサートのチケットを譲る相手を探している、といった要件をメッセージとして作成する。

#### (2) アクセス制限の設定

次に、作成した要件情報をどの範囲に配信するかを設定する。範囲は知人ネットワークにおけるリンクを基準とする。例えば、リンク 1 つ分の範囲を設定すると、検索ユーザーの友人にのみ配信される。リンク 2 つ分の範囲を設定すると、検索ユーザーの友人に加え、「友人の友人」まで配信される。本論文では、知人ネットワーク上の  $N$  リンクの範囲のことを、 $N$  クリークの距離における到達範囲と表現する ( $N=0,1,2,3,\dots$ )。

また、次に検索ユーザーは、 $N$  クリークそれぞれの到達範囲のノードに対して、伝達・開示する情報を設定する。例えば、1 クリークのノードである自身の友人に対しては、大半の情報を公開するが、2 クリークの到達範囲である「友人の友人」に対しては、氏名や E-Mail アドレス等、個人を識別する情報を開示しない、といった設定をする。このことによって、

2 クリークの受信者に対しては、送信者は自身のセンシティブな情報を限定して開示することが可能になる。2 クリークの受信者に対しては、代わりに送信者と受信者の間に存在する共通の知人に関する情報が提供される。

### (3) 要件の送信

次に、要件情報をメッセージ配信機能によって送信する。

この際、ユーザの作成した要件情報は、知人の登録によって形成されたユーザの知人ネットワークにそって、あらかじめ設定したクリーク数の到達範囲まで配信される。

### (4) 要件の受信と返信

検索ユーザからの要件情報を受信した不特定多数のユーザ（具体的には、検索ユーザが設定した  $N$  クリークの距離における到達範囲内の全てのユーザ。以下、受信ユーザ）は、伝播してきたメッセージに含まれる要件情報に対し、2 つの反応を示すことができる。受信ユーザは、送られてきた要件情報に対し、興味がなければ無視をし、興味があれば人材検索者とのランデブー確立を要求する。

この際、検索ユーザの識別情報が開示されている場合は、そのまま検索ユーザとのランデブーを確立できる。しかし、検索ユーザの設定次第では、検索ユーザの識別情報が開示されていない。したがって、本来はこのままではランデブーを確立できない。そこで本モデルでは、システムが記憶している要件情報メッセージの伝播経路（知人ネットワーク上をメッセージが配信されてきた経路）を用い、その経路を逆にたどることで、検索ユーザ自身の識別情報を用いずに検索ユーザへのランデブー確立を実現する。

なお、検索ユーザの識別情報が公開されていない場合であっても、必ず検索ユーザと受信ユーザの間に存在する共通の知人に関する識別情報が提供される。 $N > 2$  のときは、受信ユーザから 1 クリークの距離の知人の識別情報が提供される。

### (5) ランデブーの確立後

先ほど述べた手段により、検索ユーザと受信ユーザの間での 1 対 1 のランデブーを確立できる。

この際、受信ユーザ側も検索ユーザに対し、自身の識別情報を開示することも、開示しないことも可能である。つまり、設定次第では、両者共に識別情報を開示しない匿名コミュニケーションが成立する。但し、この場合における匿名コミュニケーションの特徴として、ランデブーを確立している 2 ユーザはお互いに同一のユーザとコミュニケーションを取っていることが保障される。つまり、識別情報が開示されていないことを悪用して、コミュ

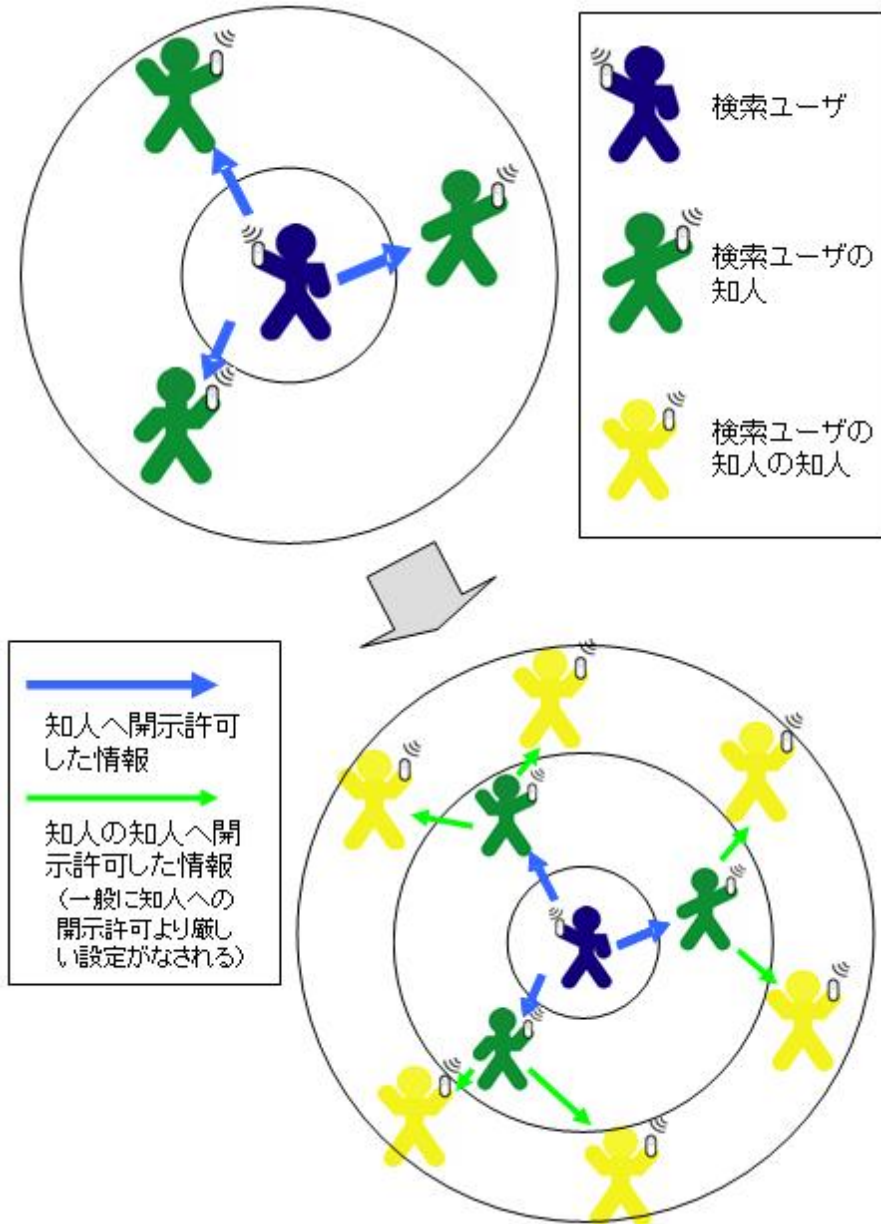


図 3.4: 要件の送信

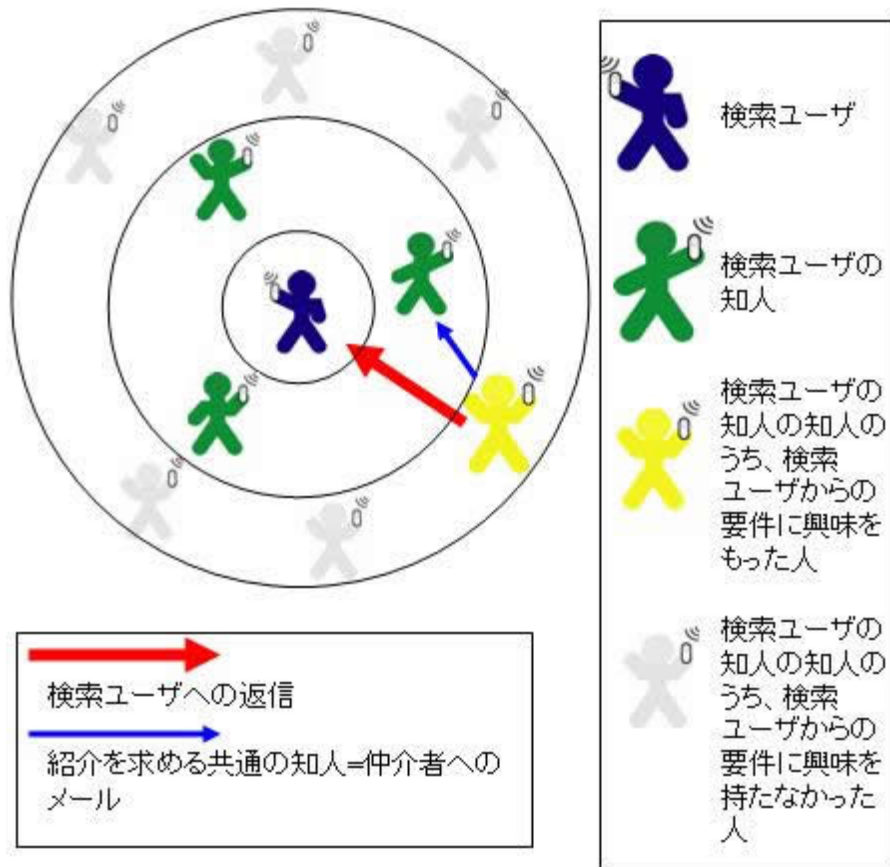


図 3.5: 要件への返信

パブリックボックス (新着: 3件)		
件名	紹介者	送信者
ITベンチャーで、ストリーミング技術者を探しています。	じゅん	非公開
誰か加藤文俊研究プロジェクトの詳細を知っている人いるー?	なし	じゅん
三菱総研にOBのお知り合いがいる方ご紹介ください		須子善彦

検索ユーザ(送信者)の識別情報が開示されていない場合でも、検索ユーザと受信ユーザの間の共通に知人の識別情報が提供される。

図 3.6: 共通の知人の識別情報



ニケーションの途中でコミュニケーション相手が別のユーザになりかわるといったことが起こらない。また、匿名コミュニケーションを行う2ユーザの距離が2クリークの場合は、2ユーザの間に必ずお互いがその存在を知り、双方向の信頼関係が構築されている知人が存在する。この知人は2ユーザにとって「共通の知人」である。この2点が、Web上の掲示板における匿名コミュニケーションとは異なる。

1対1のランデブーを確立した後は、2ユーザはお互いが人材検索における目的達成のために相応しい相手か、相応しい要件かを、それぞれ判断するために、コミュニケーションを取る。そして、お互いの合意の上で、より詳細な要件に関する情報や、自分自身（人材）に関する情報、識別情報などを交換する。

そして、最終的に2ユーザ間の協調作業が開始される。この検索段階から協調作業段階へ変化することを、本論文では「マッチした」と表現する。



図 3.7: ランデブーの確立後

### 3.3 モデルの特徴

#### 3.3.1 情報開示リスクの管理

本モデルが実現する特徴の第一点は、知人ネットワークを用いることによって、ユーザの情報開示リスクの管理をより多くユーザ自身が行える点である。このことによって、ユーザの情報開示リスクは低減すると考えられる。

第2章にて述べたように従来モデルでは一般にすべてのユーザに開示されていた情報を、本モデルでは、ユーザ間の知人関係によって構成される、知人ネットワークの信頼の結節点を用いることで、人材検索を行うユーザに対しては、人材検索時の情報開示の範囲と内容を細かく設定することを可能とする。つまり、情報開示リスクと効用（人材検索によって達成できる利益）の間でのトレードオフ上に自主選択可能な選択肢を新たに提供する。この選択肢は、前章で紹介した Microsoft 3°のように、実社会上の身近な概念をネットワー

コミュニケーションに用いることで、トレードオフ上に実用性が高く理解が容易で、実行可能な選択肢である。この選択肢によって、検索ユーザは人材検索の際、要件情報に対して関心のない第三者にまで、必要以上の検索ユーザの開示情報が伝わってしまうことを防ぎ、検索ユーザの情報開示リスクを低減する。また、受信ユーザにとっても、常に発信ユーザとの間に常に存在する「共通の知人」によって、従来の完全に Open なマッチングモデルに比べ、受信する情報に対して高い信頼性を得ることができる。なお、この点については次項にて詳しく述べる。

また受信ユーザにとっても、自身のプライバシーといったセンシティブな情報を、自身の自主選択によって保護しながら、最終的に自身の情報を開示するか否かを決定できる。また、公開するタイミングも受信ユーザ自身の自主選択によって決定できる。このような点から、受信者ユーザの情報開示リスクも低減される。

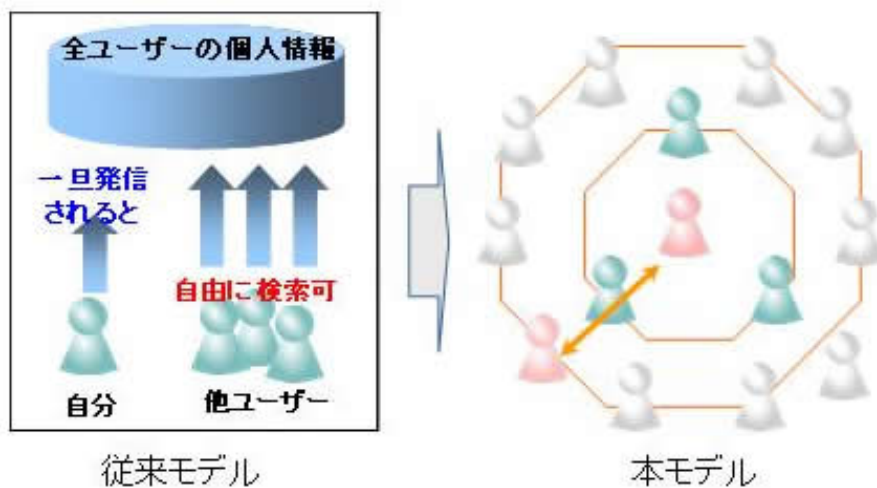


図 3.8: 情報開示リスクの管理

### 3.3.2 情報の信頼性の向上

第 2 点の有効性は、ユーザ間で交換する情報の客観的な正確性の向上である。本モデルでは、送信ユーザと受信ユーザの間に常に、両者がお互いに信頼する「共通の知人」が存在する。したがって、両者の間には、信頼の結節点が存在する。送信ユーザと受信ユーザの距離が 2 クリークである場合、両者の間には必ず「共通の知人」が存在する。

このことによって、従来のマッチングモデルでは検証が困難であったユーザのコンピテンシーの客観的正確性が検証できる。

人材検索をする際、その人材が人材検索における目的達成において適材かどうかを判断する上で、人材が保有する知識やスキル、人材のコンピテンシーを正確に把握できること

は重要である。このとき、一般的な知識やスキルに関しては、広く社会的に認知されている資格試験などの取得の有無やスコアなどによってある程度判断できる。しかしながら、特殊なスキルやコンピテンシーに関しては、資格試験など社会的権威による信頼の付与が働きにくい。コンピテンシーは、数値化といった客観的指標によって計測することが困難である。したがって、従来の人材マッチングにおいては、これらの情報は、人材自身による自己申告、自己主張を信用する以外に手段がないことが多かった。本モデルにおいては、特殊なスキルやコンピテンシーに関して、送信ユーザと受信ユーザの間に存在する「信頼の輪」によって、自己申告、自己主張の正確性について確認できる。特に、送信ユーザと受信ユーザの間が2クレークの距離である場合、両者の間には必ず「共通の知人」が存在する。多くの場合、この「共通の知人」は複数人いる。したがって、送信ユーザおよび受信ユーザは、お互いにコミュニケーション相手による自己申告、自己主張に対して、複数人の「共通の知人」に問い合わせることで、その客観的正確性について検証する手段を持つ。

このことは、従来のマッチングモデルのデメリットを回避し、マッチ率を上げることに貢献する。



図 3.9: 情報の信頼性の向上

### 3.3.3 高い実用性と広い応用範囲

#### 自律分散協調モデルによるスケーラビリティ

知人ネットワークという外部システムは、ユーザ各自が構築・活用可能なリソースであり、自律分散的な系である。ユーザ毎の構築・活用可能なリソースの量は多少ばらつきがあるかもしれないが、基本的にユーザ数の増加に対するスケーラビリティが高いと考えられる。具体的には、ユーザ毎にユーザの知人数というリソースの量が異なるが、ユーザ数の増加に応じて各ユーザが持つ知人数の平均値が変化するわけではない。つまり、本モデルはひとつの巨大なリソースプールからユーザが資源を取得するモデルとは異なり、各ユー

それぞれが自分自身に必要なリソースを自律分散的に獲得する peer to peer 型のモデルである。したがって、ユーザ数の増加によってモデルの破綻をきたすリソース不足は起こりにくく、高いスケーラビリティを持つ。

#### 自律分散協調モデルによるコスト

コストの面に関しても同様の理由に、本モデルが必要とするコミュニケーションの実現に必要なコストは小さい。本モデルは、全ユーザに対してメッセージが同報配信されるモデルではないため、全体のユーザ数が増加しても通信コストは比例増加しない。また、信頼性の維持や不正利用の監視といったコスト（それらは主に人的コストになることが多い）も小さい。本モデルでは、それらのコストは仲介者にかかる。しかし、この役割は固定的ではなく、あるときの仲介者は、次のときには仲介を受ける側になるという動的で互恵的な役割である。したがって、システム管理者がすべてのコミュニケーションを監視するといった必要は無いため、ユーザの増加に対してリニアにコストが発生しない。この点においても本モデルはスケールに対するコストパフォーマンスが良い。今日研究されている人材マッチングモデルの大半 [21][20] は、その実現のために、モデルを取り巻く多くの役割の変更やビジネスモデルの変更など、多くの社会システムの変革が必要である。それと比較して、本モデルはユーザの役割期待や必要とするスキルの敷居が低く、ビジネスモデルなど社会システムの変更が少ない。したがって、本モデルは、特に導入のしやすさといった点から高い実用性を持っている。

#### 属性マッチングの排除とランデブー型マッチング

これまで述べてきた本モデルの特徴は、本モデルが従来のマッチングモデルとは異なった設計ポリシーによって成り立っている。

2.3.1 にて述べたように、多くのマッチングモデルは、引き合わされる 2 者間のニーズの一致を、登録された属性の情報に従って判断するのが一般的である。この方法は、ユーザにシステム利用時の初期コストを与える点や、サービス提供者の想定外のユースケースに対応できない可能性、さらにマッチングの方法（アルゴリズム等）によってはミスマッチが発生する等の問題があると述べた。ここでさらに指摘したい点として、これらのモデルでは、ニーズの一致に関して、2 者間を最初に引き合わせた時点（最初のコンタクトの時点）で高い精度を実現するように設計されていることが挙げられる。

一方、本モデルの設計ポリシーは、それとは異なる。これまで述べてきたとおり、本モデルはユーザの属性に関する情報をマッチングにおいては利用せず、知人ネットワークを用いてユーザ間でメッセージをやり取りすることでマッチングを行う。

この設計ポリシーの差が生み出すものとして、いくつかの点がある。まず、ニーズの一

致はユーザが判断する。したがって、本モデルではマッチングの方法（アルゴリズム等）によるミスマッチは発生しない。また、最初のコンタクトの時点では、2者はお互いに相手に関する、或いは、要件に関する十分な情報を持ちえていないことがほとんどである。つまり、最初のコンタクトの時点では、高い精度のニーズ一致を求めている。2者は、共通の知人を介したランデブーによって、その後、各自の自己判断によって、求める情報を相手に要求し、また自身の情報を相手に開示する。このランデブーの過程を経て、お互いのニーズを一致させていくというのが、本モデルである。

したがって本モデルは、最初のコンタクトにおける高い精度を実現しない代わりに、ユーザの参加やモデル自身のシンプルさ、システムの運用コスト等を下げ、多くのユーザにマッチングサービスを提供することを可能にした。一方で、マッチングにおいては、マッチングの方法（アルゴリズム等）やニーズの一致に関するやり取りをユーザ側に委ねる事で、ユーザの自由度を高めた。このことは、従来プロのオークショナーに委ねていたオークションに対する、今日のインターネットオークションの差に似ているといえるだろう。

#### モデルの広い応用範囲

また、本モデルは第1章にて述べたように、ユーザが情報開示の相手をより適切に選択することを支援し、情報開示に関するリスクを低減することを目的として、知人関係という外部資源をコミュニケーションモデルに導入するものである。したがって、本モデルはプライバシーや知的財産、機密情報といったセンシティブな情報開示を必要とするコミュニケーションにおいて広く応用できるものである。センシティブな情報の開示を必要とするシチュエーションは今後増えてくる。その際、センシティブな情報であるゆえに、開示する行為に当たっては慎重にならなくてはならない。本モデルは、センシティブな情報の保護と信頼性が高く要求されるシチュエーションにおいて広く有効性を発揮し、その応用範囲も広い。例えば、健康や福祉、教育や人材などの分野である。健康の分野では病歴等のセンシティブ情報を用いた医師の検索、福祉の分野ではホームヘルパーやベビーシッターの検索、人材の分野ではヘッドハンティングなどに利用できる。また電子タグ技術と共にその実用化が進んできた位置情報サービスにおいて、その情報開示の管理手段として本モデルの応用が考えられる。位置情報サービスにおいて提供される情報は、コンテキストによってはユーザのセンシティブ情報となる。この情報を管理する際に、どの状況においてどの相手には開示してよい、といったアクセス制御を行う手段について、様々な研究がなされているが、ユーザにとって容易に理解できる制御手段の一つとして本モデルは有効であると考えられる。

## 第4章 Degrees Connect の設計

### 4.1 システムの全体構成

本システムは、5つのモジュールから成り立つコアファンクション層と、コアファンクション層が提供する機能をAPIにて利用し、ユーザへユーザインターフェースと共に提供するアプリケーション層によって成り立つ。

アプリケーション層は、実装するアプリケーションによって変化する。  
コアファンクション層の5つのモジュールは以下の通りである。

1. ユーザ管理機能
2. 知人リンク認識・管理機能
3. 知人ネットワーク蓄積機能（知人ネットワークデータベース）
4. 知人ネットワーク処理機能（マッチングエンジン）
5. メッセージ処理機能

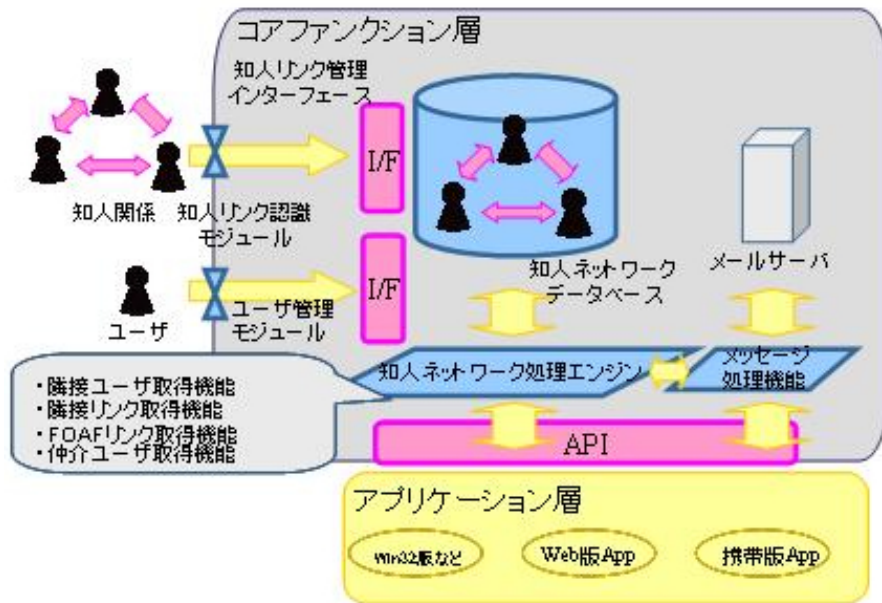


図 4.1: システムの全体構成

## 4.2 ユーザの個人情報の取得機能

コアファンクション層の第一のモジュールは、ユーザ管理モジュールである。このモジュールが提供する機能は、システムを動作させる上で必要な個人情報（識別情報、人材に関する情報、趣味・嗜好に関する情報など）を、システムを利用するユーザから取得することである。

具体的な取得方法の最適な選択肢はアプリケーションに大きく依存する。したがって本設計では、コアファンクション層の各機能が動作する上で必要最低限の個人情報の項目を定義し、その項目に対してユーザに情報を入力させるためのインターフェースを用意することで、具体的な取得方法を自由に実装できるようにした。コアファンクション層の各機能が動作する上で必要最低限の個人情報として、以下の項目を定義した。

表 4.1: ユーザから取得する個人情報項目

項目	項目の説明
ユーザの本名	実社会上でユーザを識別する識別子である姓名
システム上のユーザ識別名	システムがユーザを一意に識別する識別名
ユーザ認証用パスワード	ユーザの本人確認のためにシステムが必要とする付加情報。ユーザ本人しか知りえない文字列として利用することで本人確認を行う。
ユーザ識別子	ユーザを一意に識別可能でかつ実社会上で広く認知された識別子 (E-Mail アドレスや SSN など)
ユーザへのメッセージ送信用識別子	E-Mail アドレスなど、システムがユーザへのメッセージを送信する際に必要な識別子。ユーザ識別子がこれを兼ねていても良い。

## 4.3 ユーザの知人リンクの取得機能

### 4.3.1 ユーザの知人リンク取得に関する機能群

コアファンクション層の第二のモジュールは、知人情報取得モジュールである。このモジュールの目的は、ユーザの実社会上の情報である知人関係を、システムに取り入れる機能を提供することである。

この機能は大きく分けて 3 つの機能に分かれる。

第一機能は、ユーザ間の知人関係を認識する機能である。本システムでは、ユーザ間の知人関係のうち、知人関係にある 2 人のユーザが相互に承認した知人関係のみを知人リン



表 4.2: 知人リンク取得に関する機能群

第一機能	知人関係の認識機能
第二機能	知人関係の承認機能
第三機能	知人リンクの構築機能

クとして扱う。この知人リンクのみが本システムにおける知人ネットワークを構成するリンクとなる。すなわち、実社会上で知人関係であったとしても、ユーザによる相互の承認といった手続きを踏まない限り、本システムでの人材マッチングに利用される知人リンクとはならない。この際、ユーザの手続きを簡素化するために、コンピュータシステムが認識可能な事象のうち、実社会上の知人関係と解釈できるものを認識し、ユーザに候補として提供するのが第一機能である。

そして、その候補を、実際にユーザの主観的判断と操作によって承認あるいは拒否させる機能を提供するのが第二の機能である。そして、第二機能によって承認された場合に、実際に本システム上の知人ネットワークを構成する知人リンクを構築するのが第三機能である。

#### 4.3.2 知人関係の認識機能

第一機能の知人関係認識機能について述べる。この機能の具体的な実現方法は、いくつか考えられる。したがって、アプリケーションやアプリケーションの動作環境によって、容易に複数の選択肢を選択できることが必要である。なお、具体的な実現方法の選択肢は、以下の手段等が考えられる。

##### (1) ディレクトリーサービス

LDAP[53][54]等のディレクトリーサービスの情報を利用・応用して、ユーザの識別と、ユーザが保持する知人関係を取得する方法である。この方法は、主に本システムが一つの組織内で使われる場合に有効である。ディレクトリーサービスの実装によっては本システムのために拡張をする必要が生じる点や、運用に関して新たに本システムのためにルールを作る必要がある点がデメリットである。

##### (2) インスタントメッセンジャ

ICQ[55], AOL Messenger, MSN Messenger, Jabber[56]といったインスタントメッセンジャの機能を利用して、ユーザの識別と、ユーザが保持する知人関係を取得する方法であ

る。この方法は、インスタントメッセージにユーザの識別機能やユーザが保持する知人関係を取得する機能が備わっているために実装が容易な点や、インスタントメッセージにはユーザ間でメッセージを送受信する機能も備わっている点で、本システムの他の機能を実現する上でも親和性が高い。また、インスタントメッセージ自体は既に多くのユーザが利用しており、かつ基本的には World Wide でオープンなプラットフォームである点で、多くのユーザに開かれた選択肢である。

一方で現状では、インスタントメッセージの機能を利用するための API は、インスタントメッセージの規格によって異なる点や、規格によっては非公開な部分が存在する点、規格が異なるインスタントメッセージ間の通信が困難である点がデメリットである。

### (3) Web アプリケーション

前述のインスタントメッセージのデメリットを克服する方法として、Web アプリケーションに独自のインスタントメッセージング機能を実現する手段が考えられる。Web アプリケーション上に、ユーザ毎にバディリストを作成する機能を提供する。

この手段では、Web アプリケーションを新たに利用してもらう必要がある点がデメリットであるが、閉じられた組織内での利用においてはイントラネットという形で、Open なコミュニティにおいても、ASP という形で提供できる。

### (4) メールのパース

本システムを利用する組織・コミュニティのメールサーバーに特別なパーサーを設置し、メールの情報をういて、ユーザ間の知人関係を推測し、取得する方法である。2 にて紹介した ANS[46] もこの方法を用いている。この方法は、ユーザの知人関係の取得を自動化する点で便利であるが、ユーザがやりとりするメールの機密性が侵害される点や、取得される知人リンクの精度が、本システムの目的に対して、低すぎる可能性がある。

### (5) Web リンク、トラックバック等

ユーザの保持する Web 間のリンクやトラックバックの存在によって、ユーザ間の人脈リンクを取得する方法である。

この方法は、ユーザの人脈リンク取得の多くのステップを自動化する点で便利であるが、この方法も先ほどのメールのパースと同様に、取得される知人リンクの精度に問題がある可能性が高い。また、実際には、ユーザが各自の Web サイト等をシステムに記録する必要があるため、自動化できるステップは限られる。また、ユーザが自身の Web を持っていない場合もあり、この手段だけを選択するのは少々困難である。したがって、この手段は他

の手段の前準備のオプションとして組み合わせて利用することが考えられる。

## (6) FOAF

FOAF[57] とは、人々に関する情報や人と人とのつながりを公開、共有するため、the foaf project が進める RDF/XML[58][59][60] を使った仕組みである。ユーザ各自が FOAF の DTD で決められた項目に従って個人情報を提供する、電子的な名刺のようなものである。その項目の一つに、知人の FOAF を記述する項目がある。この項目を読み取ることで、ユーザ間の知人関係を取得できる。この方法は、一度 FOAF を作成してしまえば、ユーザの個人情報と知人リンクの両方を自動的に取得できる。しかし、FOAF が現時点で広く認知されていない点が問題である。

上記に述べた例をはじめ複数の実現方法に対応するため、具体的な実現方法に依存せず知人リンクを取得することを可能とするインターフェースを定義した。

表 4.3: 知人リンク管理インターフェース

知人リンク管理インターフェース
コマンド (構築、解除)
始点ユーザ識別クラス
終点ユーザ識別クラス

### 4.3.3 知人関係の承認機能

前節で述べたユーザ間の知人関係の認識機能で取得した知人リンクの「候補」は、リンクの始点と終点の 2 ユーザの相互承認を経て、本システムに有効な知人リンクとなる。

2 ユーザの相互承認を経た、本システムに有効な知人リンクを、「双方向の信頼関係」をもった知人関係と以下表現する。

ユーザは「双方向の信頼関係」をもった知人リンクを確立するために、知人リンクの「候補」のうち実際に知人リンクを構築したい相手に対して、承認を求める処理を行う。この処理を行うのが知人関係の承認機能である。

ユーザは人脈リンクを構築したい相手に対して承認を求めるとき、知人リンクの構築要求メッセージを相手に通知する。この際、システムには知人リンク構築要求レコードという名の有向リンクが生成され、一時的に記録される。この知人リンク構築要求レコードは、表 4.4 に示したリンクとして定義され、システムに一時的に保存される。この際、リンクの始点のユーザ (知人リンク構築を要求したユーザ) から、終点のユーザ (知人リンク構築

を要求されたユーザ)へ、知人リンクを構築するための必要な承認を求める通知メッセージが届く。終点のユーザは、この通知メッセージに記載されている指示に従って操作することで、承認もしくは拒否をすることができる。また、何も操作をしないという形で拒否をすることも可能である。

なお、ユーザは知人リンク構築要求を行った時点で、自身のバディリストに、承諾待ち状態として登録される。構築要求を送られた相手が明示的に拒否をしない限り、相手の名前は承諾待ち状態のままリストに登録される。

表 4.4: 知人リンク構築要求レコード

始点ユーザ識別子	知人リンク構築を要求したユーザをシステムが一意に識別する識別子
終点ユーザ識別子	知人リンク構築を要求されたユーザをシステムが一意に識別する識別子
通知メッセージ	始点ユーザから終点ユーザへの通知メッセージ
終点ユーザへのメッセージ送信用識別子	終点ユーザへ通知メッセージを送るためのメッセージ送信用識別子 (多くの場合は、E-Mail アドレス)
リンク構築要求識別子	リンク構築要求一つ一つにつけられる任意の識別子
要求日時	リンク構築要求の日時を示すタイムスタンプ

承認もしくは拒否をすることで、知人リンク構築要求レコードはシステムから削除され、かつ、ユーザは知人リンク構築要求を行ったユーザのバディリストからも削除される。

知人リンク構築要求が承認されると、知人リンク構築要求レコードの情報は次項で説明する第三機能の「知人リンクの構築機能」に引き渡される。

#### 4.3.4 知人リンクの構築機能

前節で述べた機能により、リンクの始点と終点の2ユーザの相互承認を経て、知人リンクの形成要求が承認された知人形成要求リンクは、本システムにおける知人関係の成立条件である「双方向の信頼関係」をもった知人とみなされ、知人リンクが形成される。

知人リンクは、システム上では双方向の知人リンク (正確には、方向の異なる2本の片方向の知人リンク) によって表される。知人リンクは、表 4.5 の要素で示される。

表 4.5: 知人リンクの構成要素

始点ユーザ識別子	知人リンクの始点ユーザをシステムが一意に識別する識別子
終点ユーザ識別子	知人リンクの終点ユーザをシステムが一意に識別する識別子
終点ユーザへのメッセージ送信用識別子	終点ユーザへ通知メッセージを送るためのメッセージ送信用識別子（多くの場合は、E-Mail アドレス） 但し、システムが終点ユーザ識別子を用いることで、メッセージ送信用識別子を呼び出すことができれば、本項目は不要

#### 4.3.5 知人リンクの編集機能

知人リンクを削除する場合は、前項で述べた知人リンクを示すオブジェクトを削除する。この際、双方向リンクのうち、片方のリンクが削除されるため、本システムにおける知人リンクの定義である「双方向の信頼関係」が解除される。したがって、その時点で、本システムにおける知人リンクではなくなる。

この削除操作は、削除行為を行った相手ユーザに対して、削除した旨を通知するオプションに加え、相手ユーザのバディリストには何の変化も起こさないオプションを選択できる。後者のオプションは、実社会上のユーザ間の人間関係に対する影響を低減するためである。なお、どちらのオプションを選択しても、削除操作を行った時点で削除行為を行ったユーザのバディリスト上およびシステム上での知人リンクは削除される。

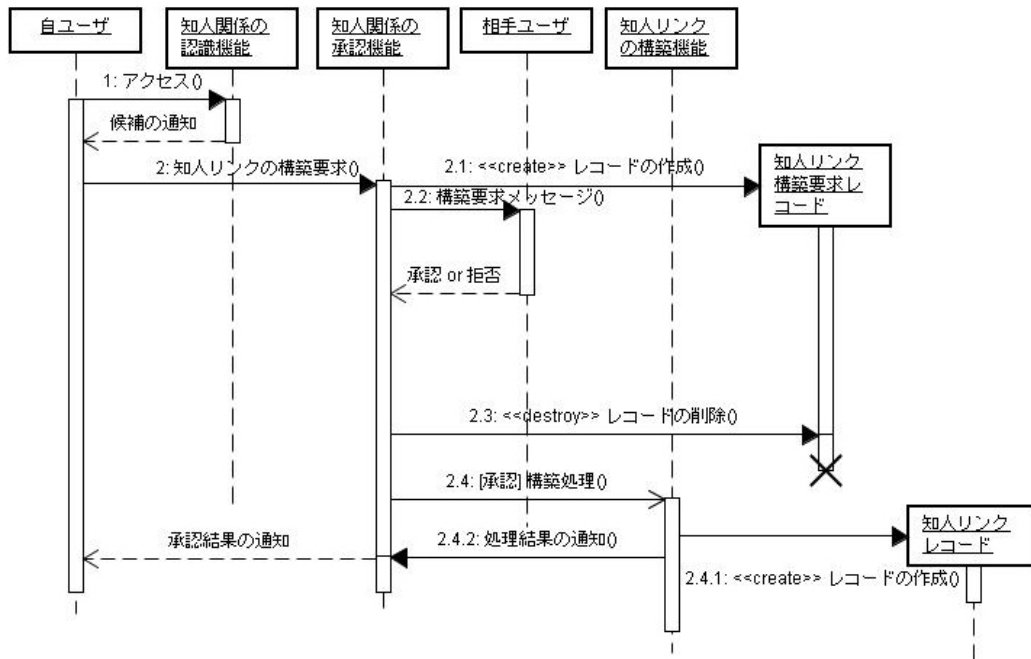


図 4.2: 知人リンクの構築プロセス

#### 4.4 知人リンクの蓄積機能

知人リンクは前節で述べたように、表 4.5 に示されるオブジェクトである。システムはこのオブジェクトを永続的に保持・蓄積する必要がある。この提供するのが知人リンクの蓄積機能である。

知人リンクオブジェクトは、リレーショナルデータベースのレコードとして蓄積されることを想定して設計している。ただし具体的な蓄積方法については実装依存である。また、保持する場所（リレーショナルデータベースの場合、このレコードを保持するテーブルの格納場所）は、ユーザ毎に個別に用意することも、一箇所に保持することも可能である。

しかし、いかなる実装方法を用いても、後に述べる知人ネットワーク処理機能を実現するために定義される API を通じて、蓄積されたデータへアクセスできることが求められる。

#### 4.5 知人ネットワーク処理機能

知人ネットワーク処理機能は、本システムの最も重要な機能である人材マッチングのための諸機能を提供する。

本機能は、知人ネットワークを処理し、ユーザ間のコミュニケーションを実現するために以下の機能を API 経由でアプリケーション層に提供する。

表 4.6: 知人ネットワーク処理に関する主要な機能

隣接ユーザ取得機能	指定されたユーザから 1 クリークの距離の到達範囲にあるユーザの識別子リストを取得
隣接リンク取得機能	指定されたユーザから 1 クリークの距離の到達範囲にあるユーザとの知人関係を示すリンクオブジェクトのリストを取得
FOAF リンク取得機能	指定されたユーザから N クリークの距離の到達範囲にあるユーザとの知人関係を示すリンクオブジェクトのリストを取得
仲介ユーザ取得機能	指定した 2 ユーザの間に存在するユーザのリストを取得

## 4.6 知人ネットワークを用いたメッセージ処理機能

知人ネットワークを用いたメッセージ処理機能は、前節で説明した知人ネットワーク処理機能と連携し、人材マッチングのために必要なユーザ間のコミュニケーションを実現するための機能を API 経由でアプリケーション層に提供する。具体的には以下の機能である。

表 4.7: 知人ネットワークを用いたメッセージ送信機能

隣接ノードへのメール送信機能
2 クリークの距離へのメール送信機能
N クリークの距離へのメール送信機能

## 第5章 Degrees Connectの実装

### 5.1 実証ターゲットと実装ポリシー

コアファンクションは、Java 言語 (JDK version 1.4.1)[61] を用いて実装した。コアファンクションの API は、Java のメソッド呼び出しでアプリケーションから利用する。したがって、現在はアプリケーションも Java で実装した。

本研究でのマッチングモデルの有効性を実証するターゲットとして、学会イベント、同窓会イベントと同窓会サイトの3つを設定した。アプリケーションは、それらターゲットに合わせ、以下の2種類の実装を用意した。

表 5.1: 実装の種類と利用するターゲット

実装の種類	利用するターゲット
Web サイト版アプリケーション実装	Web サイトなど継続性が求められるターゲット
携帯電話版アプリケーション実装	イベントなど即時性が問われるターゲット

Web アプリケーション実装は、コミュニティサイトに埋め込まれる形で統合される実装で、J2EE[62] を用いて実装した。共通の目的や興味を持ったユーザ間で利用されるコミュニティサイトでの使用が想定される。

携帯電話インターフェース実装は、携帯電話からメールと Web アクセスによって利用される実装で、J2EE を用いて実装した。携帯電話インターフェース実装は、イベントなどの時限的でリアルタイム性が重要視される状況での使用を想定している。

### 5.2 実装環境

前節で述べた2つの実装は、以下の環境で実装を行った。

また、実装は主に、コアファンクション層とアプリケーション層の間の結合を疎結合にすることによって、アプリケーションの実装環境に自由度を持たせた。具体的には、J2EE デザインパターンのレイヤリング設計を参考に、プレゼンテーション (Web) 層とファンク



表 5.2: 実装環境

環境の項目名	スペック等
マシンアーキテクチャ	PC-AT 互換機
CPU	PentiumIII 800MHZ * 2
メモリ	512MB
OS	Red Hat Linux7.3 (Kernel 2.4.20)
J2SE SDK	Sun J2SE SDK 1.4.2
Servlet Container	Jakarta Tomcat 4.1.29 (Servlet API 2.3 / JSP 1.2)
RDBMS	PostgreSQL 7.1.3 + JDBC Driver

ション (ロジック) 層、データアクセス層の 3 層に機能を分割した。

また、今回用意した 2 つの実装では、Web 層において Jakarta Struts 1.01 フレームワーク [63] を利用し、ファンクション (ロジック) 層、データアクセス層は POJO によって実装した。Struts フレームワークにおける実装は、View の実装に JSP1.2 と Struts タグライブラリ、JSTL[64] を用いた。Model の実装は、POJO を用いた。その際、ファンクション (ロジック) 層を 2 つのレイヤーにわけ、プレゼンテーション (Web) 層よりアプリケーション層から呼び出す高レベル API を実現する層を、データアクセス層よりコアファンクション層を実現する低レベル API を実装した。

高レベル API を呼び出すアプリケーション層の実装を変更することで、Win32 アプリケーションといった他の様々なアプリケーション形態の実装を容易とした。

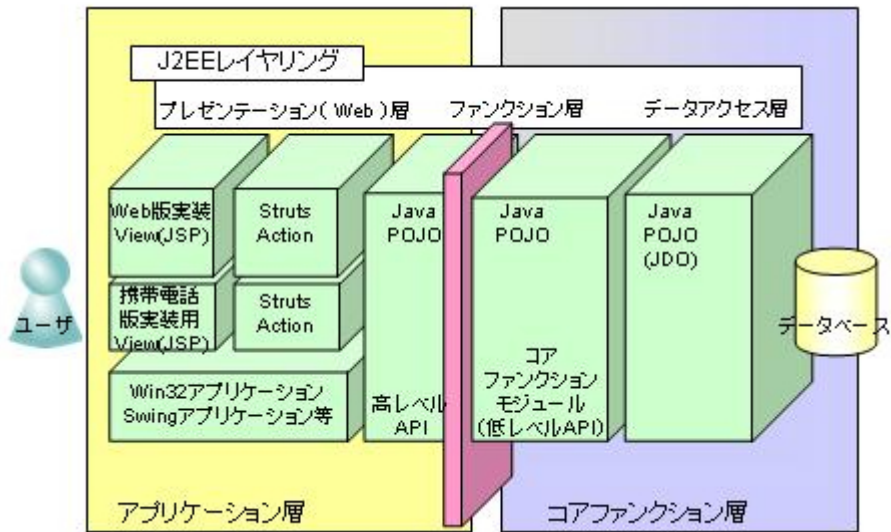


図 5.1: 実装の全体像とレイヤリング

### 5.3 知人ネットワークの構成要素の実装

ユーザの個人情報の取得・管理の機能を提供するモジュールの説明に先立ち、本システムの知人ネットワークの構成要素の実装について説明する。本システムでは、ユーザは知人ネットワークのノードとして扱われる。また、ユーザ間の知人関係はリンクとして表現される。

それぞれは、以下のインターフェースによって規定される。ユーザ情報を保持する全てのオブジェクトは、表 5.3 に示す Node インターフェースを実装する。また、全ての知人関係を示すオブジェクトは、表 5.4 に示す Link インターフェースを実装する。

表 5.3: core.framework.network.Node インターフェース

<<interface>> core.framework.network.Node	
<b>実装が想定されるフィールド</b>	
- java.lang.String nodeId	ノード識別子
- boolean activated	ノードが有効かどうかを示すフラグ
<b>メソッド</b>	
+ java.lang.String getId()	ノード識別子を取得
+ int getIdByInt()	可能ならばノード識別子を int 型で取得
+ void init()	ノードの初期化処理を行う
+ boolean isActivated()	ノードが有効かどうかを取得
+ void setActivated(boolean)	ノードの有効フラグを設定
+ void setId(int i)	ノード識別子を設定
+ void setId(java.lang.String)	ノード識別子を設定
<b>既知の実装クラス</b>	
core.framework.network.AbstractNode	

前述した各インターフェースの実装クラスについては、次節以降で説明する。

表 5.4: core.framework.network.Link インターフェース

<<interface>> core.framework.network.Link	
<b>実装が想定されるフィールド</b>	
- boolean activated	リンクが有効かどうかを示すフラグ
- boolean direction	有向リンクであるかどうかを示すフラグ
- Node startNode	リンクの始点を示すノード
- Node endNode	リンクの終点を示すノード
<b>メソッド</b>	
+ Node getStarNode ()	始点のノードオブジェクトを取得
+ java.lang.String getStarNodeId()	始点のノード識別子を取得
+ Node getEndNode ()	終点のノードオブジェクトを取得
+ java.lang.String getEndNodeId()	終点のノード識別子を取得
+ boolean hasDirection()	有向リンクであるかどうかを取得
+ boolean isActivated()	リンクが有効かどうかを取得
+ void setActivated(boolean)	ノードの有効フラグを設定
+ void setDirection(boolean)	ノードの有向フラグを設定
+ void setStartNode(Node)	始点ノードを設定
+ void setEndNode(Node)	終点ノードを設定
<b>既知の実装クラス</b>	
core.framework.network.AbstractLink	

## 5.4 ユーザの個人情報管理に関する機能

### 5.4.1 ユーザの個人情報の処理

ユーザの個人情報は、表 5.5 に示すとおり、大きく 2 つに分けられる。

表 5.5: ユーザの個人情報の分類

項目	説明
必須項目	システムが動作する上で必要最低限の項目
任意項目	システムが付加機能を提供する上で必要な項目

このうち、必須項目は、4.2 および表 4.1 で示した項目を実装するものである。本システムではユーザを示す全てのオブジェクトは前節で述べた Node インターフェースを実装する必要がある。また、さらに必須項目の個人情報オブジェクトは、Node インターフェースに加え、表 4.1 で示した項目を示すインターフェースである

`core.framework.foundation.DCUserCert` インターフェースを実装する。

また、本システムでは、ユーザの認証および、HTTP セッションの維持のために、必須項目の個人情報オブジェクトを用いる。したがって、`DCUserCert` インターフェースには、ユーザ認証のためのメソッドも定義されている。

本システムでは、ユーザにログイン処理を行わせ、ユーザからユーザ識別子とユーザ認証用パスワードを入力させる。その入力データを、`DCUserCert` インターフェースの実装クラス `DCUserCertBeans` (詳細は後述) 或いは、そのサブクラスによって取得し、メソッド `UserCertify()` を用いて、ユーザ認証を行う。ユーザ認証が成功した場合、`DCUserCert` インターフェースの実装クラスのインスタンスを `session` スコープに格納し、各種ページやモジュールが利用する。ユーザの明示的なログアウト処理、或いは、ユーザが HTTP クライアントを終了させた時点、あらかじめ設定したセッション保持時間が経過した時点で、インスタンスは削除され、ユーザのセッションは終了する。

`DCUserCert` インターフェースの既知の実装クラスとして、`DCUserCertBeans` を実装した。この実装クラスは、`DCUserCert` インターフェースのすべてのメソッドを実装した上で、Degrees Connect の全てのアプリケーションにおいて共通に利用すると考えられる任意項目をプロパティに持つ `JavaBeans` である。

本システムは、Web 層のフレームワークとして Jakarta プロジェクトの `Struts1.01` を用いて実装した。したがって、`DCUserCertBeans` は、`Struts` の `Action Form` クラスを継承する。

表 5.6: core.framework.foundation.DCUserCert インターフェース

<<interface>> core.framework.foundation.DCUserCert	
<b>実装が想定されるフィールド</b>	
- java.lang.String password	ユーザ認証用パスワード
- java.lang.String user_id	システム上の識別名
- java.lang.String user_name	ユーザ識別子
- java.lang.String com_id	ユーザへのメッセージ送信用識別子
<b>メソッド</b>	
+ java.lang.String getPassword()	ユーザ認証用パスワードを取得
+ void setPassword(String)	ユーザ認証用パスワードを設定
+ java.lang.String getUser_id()	システム上の識別名を取得
+ void setUser_id(String)	システム上の識別名を設定
+ java.lang.String getUser_name()	ユーザ識別子を取得
+ void setUser_name(String)	ユーザ識別子を設定
+ java.lang.String getCom_id()	ユーザへのメッセージ送信用識別子 を取得
+ void setCom_id(String)	ユーザへのメッセージ送信用識別子 を設定
+ boolean UserCertify()	ユーザ認証を行うメソッド
<b>既知の実装クラス</b>	
core.logic.foundation.DCUserCertBeans	

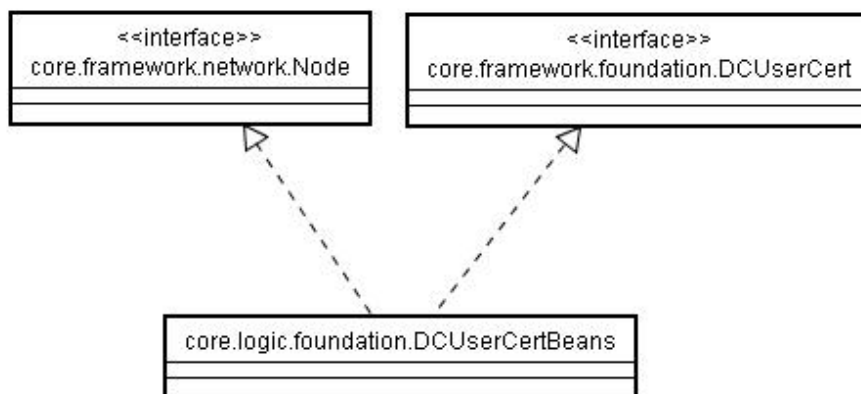


図 5.2: ユーザの個人情報クラス群の関係

### 5.4.2 ユーザの個人情報の取得

ユーザの個人情報に対する大半の処理は、抽象クラスのファクトリークラス `core.framework.foundation.UserUtilsFactory` の実装クラスの `getInstance()` メソッドによって生成されるユーザ情報管理ユーティリティクラスのインスタンスによって行う。(Factory Method パターン)

ユーザ情報管理ユーティリティクラスは、`core.framework.foundation.UserUtils` インターフェースの実装クラスであり、デフォルトの実装クラスは

`core.logic.basic.DCUserUtils` クラス (Singleton パターン) である。

ユーザの個人情報取得も本ユーティリティクラスが提供するユーザ登録用 API (具体的には、`UserUtils::registerUserWithBean(UserCert)`) を用いる。

本研究でのアプリケーション実装では、ユーザは Web サイトでユーザ登録を行う。その際に、ユーザはアプリケーションに必要な個人情報 (任意項目も含む) を記入し、Submit ボタンを押すことで、個人情報が HTTP プロトコルによってサーバに送信され、Struts の Action 処理を経由して、本ユーティリティクラスが提供するユーザ登録用 API が呼び出され、ユーザの個人情報がシステムに取得される。

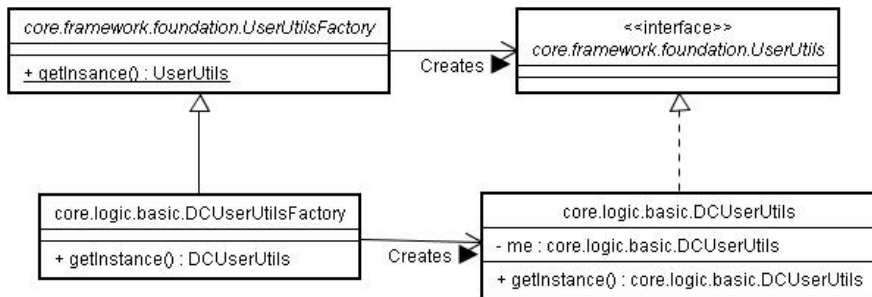


図 5.3: ユーザ情報管理ユーティリティクラス群の関係

表 5.7: `core.framework.foundation.UserUtilsFactory` 抽象クラス

<<abstract class>> core.framework.foundation.UserUtilsFactory	
メソッド	
- static UserUtils getInstance()	UserUtils の実装クラスのインスタンスを返却
既知の実装クラス	
- core.logic.basic.DCUserUtilsFactory	

表 5.8: core.framework.foundation.UserUtils インターフェースの一部

<<interface>> core.framework.foundation.UserUtils	
実装が想定されるフィールド	
- DCUserUtils me	Singleton 用フィールド
メソッド	
+ DCUserUtils getInstance()	Singleton インスタンスの取得
+ UserCert registerUserWithBean(UserCert)	ユーザ登録メソッド
(他のメソッドは省略)	

## 5.5 ユーザの知人リンクの取得機能

### 5.5.1 ユーザの知人リンクの管理機能群

本節では、コアファンクション層の第二のモジュールである、知人情報取得モジュールの実装を説明する。まずは、その中心的な機能を提供するネットワーク管理ユーティリティクラスの説明をする。

知人ネットワークに対する大半の処理は、抽象クラスのファクトリークラス core.framework.network.NetworkUtilsFactory の実装クラスの getInstance() メソッドによって生成されるネットワーク管理ユーティリティクラスのインスタンスによって行う。(Factory Method パターン)

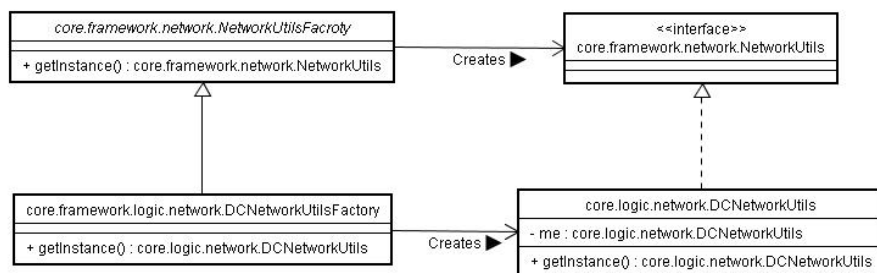


図 5.4: ネットワーク管理ユーティリティクラス群の関係

ネットワーク管理ユーティリティクラスは、core.framework.network.NetworkUtils インターフェースの実装クラスであり、デフォルトの実装クラスは core.logic.network.DCNetworkUtils クラス (Singleton パターン) である。



表 5.9: core.framework.network.NetworkUtilsFactory 抽象クラス

<<abstract class>> core.framework.network.NetworkUtilsFactory	
メソッド	
+ NetworkUtils getInstance()	Singleton インスタンスの取得
既知の実装クラス	
core.framework.logic.network.DCNetworkUtilsFactory	

表 5.9 は、core.framework.network.NetworkUtilsFactory 抽象クラスの主要なメソッドである。他には、将来的な機能拡張として、知人リンクに関係の種類や重み付けを付加するためのメソッド等が存在するが、現在は利用されていないため省略する。デフォルトの実装クラスである core.logic.network.DCNetworkUtils は、これらのメソッドを実装する。

### 5.5.2 ユーザの知人リンク取得の認識機能

ユーザ間の知人関係である知人リンクを認識する機能の実装を説明する。つまり、コンピュータシステムが認識可能な事象のうち、実社会上の知人関係と解釈できるものから、知人関係になりうる「候補の」リンクを取得する機能である。この機能は、前章で議論したとおり、アプリケーションやアプリケーションの動作環境によって具体的な実現方法を選択できることが必要である。

本システムでは、複数の実現方法に対応するため、具体的な実現方法に依存せずに知人リンクを取得するためのインターフェースを定義 (前節で説明したネットワーク管理ユーティリティ NetworkUtils:: requestUniDirectionLink(Node startNode, Node endNode)) し、実装した。

また、本研究におけるアプリケーションから、本機能の具体的な実現方法は、ユーザのプライバシーといったセンシティブな情報を侵害せず現時点で容易に実現できる点を重視し、Web アプリケーションに独自のインスタントメッセージング機能を実現する方法を選択した。

本システムが実現するモデルは、前述したように、情報配信の範囲の制限およびコミュニケーション相手の選定における信頼判断をユーザの知人に委譲するモデルである。したがって、ユーザは、信頼判断を委譲しうる信頼関係を持つ人物を知人として登録することが要求されている。ユーザは、本システムを利用する全ユーザの一覧から、登録したい知人を選び、知人リンク構築要求を出す。

表 5.10: core.framework.network.NetworkUtils インターフェースの主要なメソッド

<<interface>> core.framework.network.NetworkUtils	
<b>実装が想定されるフィールド</b>	
- NetworkUtils me	Singleton 用フィールド
<b>メソッド</b>	
+ NetworkUtils getInstance()	Singleton インスタンスの取得
+ void requestUniDirectionLink(Node startNode, Node endNode)	片方向知人リンクの構築要求を作成する
+ void makeUniDirectionLink(Node startNode, Node endNode)	片方向知人リンクを構築する
+ void makeBiDirectionLink(Node startNode, Node endNode)	双方向知人リンクを構築する
+void removeUniDirectionLink(Node startNode, Node endNode)	片方向知人リンクを削除する
+ void removeBiDirectionLink(Node startNode, Node endNode)	双方向知人リンクを削除する
+boolean checkUniLink(Node startNode, Node endNode)	片方向知人リンクの存在を検出する
+ boolean checkBiLink(Node startNode, Node endNode)	双方向知人リンクの存在を検出する
<b>既知の実装クラスの一覧</b>	
core.logic.network.DCNetworkUtils	

ユーザが知人リンク構築要求を出すことにより、知人リンク構築要求をされる相手に対して、知人リンク構築要求の承認を求めメッセージが送信される。また、システムが知人リンク構築要求の承認待ちであることを認識するために、知人リンク構築要求メッセージが生成され、記録される。知人リンク構築要求メッセージは、関係データベースの 1 レコードとして表され、friend\_auth という名前のテーブルに記録される。

このメッセージの送信先は、終点ユーザへのメッセージ送信用識別子が利用される。

表 5.11: 知人リンク構築要求テーブル friend\_auth のスキーマ

属性の説明	属性名	データ型	制限等
始点ユーザ識別子	from_id	可変長文字列	NOT NULL
終点ユーザ識別子	to_id	可変長文字列	NOT NULL
通知メッセージ	comment	可変長文字列	特になし
終点ユーザへの メッセージ送信用 識別子	com_addr	可変長文字列	NOT NULL
リンク構築要求識 別子	auth_key	可変長文字列	NOT NULL
要求日時	create_timestamp	TIMESTAMP	デフォルト値::現在時刻
有効フラグ	available	BOOL	NOT NULL, デフォルト 値::true

主キー : from\_id, to\_id の複合キー

インデックス : from\_id, to\_id

friend\_auth テーブルに追加された知人リンク構築要求レコードは、構築要求を送られた相手ユーザが、承認か拒否の操作を行うまでテーブルに保存される。通知メッセージである属性 comment は、構築要求を送るユーザが、構築要求を送られる相手ユーザに対して、要求理由などを記述した文章である。また、リンク構築要求識別子である属性 auth\_key は、主キー from\_id,to\_id と、要求が行われた時刻に、ランダムな演算を加えた文字列に対し、MD5 でハッシュ値を求めたものである。このハッシュ値は、知人リンク構築要求の承認を求めメッセージの中に、URL の一部として埋め込まれる。構築要求を送られる相手ユーザはこの URL を使って、構築要求に対して、承認或いは拒否の操作を行う。なお、本実装で用いた RDBMS は、Postgres[65] であるため、スキーマ表の可変文字列は TEXT 型のことを示す。

### 5.5.3 知人リンクの構築機能

前節で述べた機能により、リンクの始点と終点の 2 ユーザの相互承認を経て、知人リンクの形成要求が承認された知人リンクの形成要求リンクは、本システムにおける知人関係の成立条件である「双方向の信頼関係」をもった知人関係とみなされ、知人リンクが形成される。

知人リンクも RDBMS のテーブル friends にレコードとして表現される。

表 5.12: 知人リンクテーブル friends のスキーマ

属性の説明	属性名	データ型	制限等
始点ユーザ識別子	from_id	可変長文字列	NOT NULL
終点ユーザ識別子	to_id	可変長文字列	NOT NULL
更新時刻	update_time	TIMESTAMP	デフォルト値::現在時刻
論理的距離 (将来の拡張用)	distance	REAL	デフォルト値::1.00
リンクの重み付け (将来の拡張用)	weightiness	REAL	デフォルト値::10.0
有効フラグ	available	BOOL	NOT NULL, デフォルト値::true

主キー : from\_id, to\_id の複合キー

インデックス : from\_id, to\_id

リンクの構築は、前述したネットワーク管理ユーティリティクラスの makeUniDirectionLink(Node startNode, Node endNode) メソッド及び makeBiDirectionLink(Node startNode, Node endNode) メソッドを用いる。これらのメソッドの処理により、データベース上を検査し、重複が発生しないかチェックした上で、新しいレコードが追加される。

## 5.6 知人ネットワークのネットワーク演算

知人ネットワークのネットワーク演算について説明する。ネットワーク演算とは、ノードとリンクの情報を処理し、あるノードからの任意の距離のノードのリストを取得するといった機能である。本システムでは、それらの機能は、`core.framework.engine.DCMatchingEngine` インターフェースで定義されている。

従って、実際のネットワーク演算を行うクラスは、`core.framework.engine.DCMatchingEngine` インターフェースの実装クラスである。この実装クラスを変更することで、ネットワーク演算処理の詳細を変更することができる。例えば、現在 2 つのデフォルト実装があるが、

表 5.13: core.framework.engine.DCMatchingEngine インターフェース

<<interface>> core.framework.engine.DCMatchingEngine	
<b>実装が想定されるフィールド</b>	
- DCMatchingEngine me	Singleton 用フィールド
<b>メソッド</b>	
+ DCMatchingEngine getInstance()	Singleton インスタンスの取得
+ List getCouple(Node startNode)	知人のリストを取得
+ List getNeighbors(Node startNode)	知人へのリンクを示すオブジェクトを取得
+ List getFOAF(Node startNode, List linkPipes)	知人の知人へのリンクを示すオブジェクトを取得
+ List getHubNodes(Node, Node)	2つのノードの間のノードの一覧を取得
<b>既知の実装クラスの一覧</b>	
core.logic.engine.DConnectLogic, core.logic.engine.DConnectUDLogic	

core.logic.engine.DConnectLogic 実装クラスは、モデルの設計に忠実に従った実装であるに対して、core.logic.engine.DConnectUDLogic は、知人リンクの解釈を緩め、片方向の知人リンクによって、2 ユーザ間の知人関係を処理する実装である。このように、目的に応じて、ネットワーク演算の処理の詳細を変更することが容易である。

core.framework.engine.DCMatchingEngine インターフェースの実装クラスは、core.framework.engine.MatchingEngineFactory ファクトリークラスの static メソッドを用いて生成・取得する (Factory Method パターン)

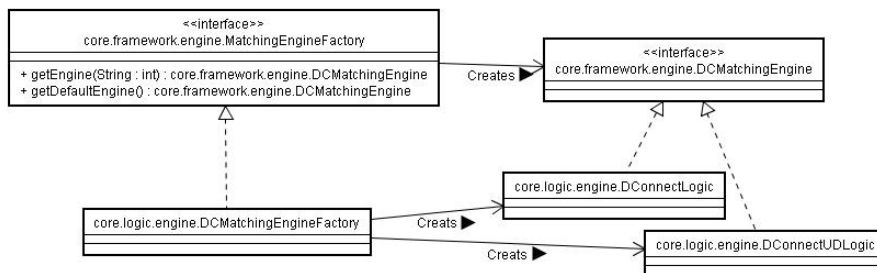


図 5.5: ネットワーク演算処理クラス群の関係

core.framework.engine.DCMatchingEngine インターフェースの実装クラス core.logic.engine.DConnectLogic の主なメソッドを説明する。

表 5.14: `core.framework.engine.MatchingEngineFactory` インターフェース

<<interface>> <code>core.framework.engine.MatchingEngineFactory</code>	
メソッド	
+ <code>DCMatchingEngine getEngine(String)</code>	引数の文字列をキーとして、JNDI から実装クラスを取得
+ <code>DCMatchingEngine getDefaultEngine()</code>	JNDI に登録されたデフォルトの実装クラスを取得

**public List getCouple(Node startNode) メソッド**

引数で渡されるノードオブジェクト(実際は、ユーザ識別オブジェクトとなる)を始点とし、相互の知人リンクを持つユーザ識別オブジェクトを `Node` 型のリストとして、`java.util.List` インターフェース型のオブジェクトとして返す。この `List` インターフェースの実装型は、`java.util.ArrayList` である。

**public List getNeighbors(Node startNode) メソッド**

引数で指定したノードオブジェクト(実際は、ユーザ識別オブジェクトとなる)を始点とし、双方向の知人リンクを持つユーザとのリンクを示す方向つきパイプオブジェクトを格納したリストとして、`java.util.List` インターフェース型のオブジェクトとして返す。この `List` インターフェースの実装型は、`java.util.ArrayList` である。

方向つきパイプオブジェクトは、`core.framework.network.Link` インターフェースの実装クラスでありデフォルトの実装クラスは、`core.logic.network.DConnectLinkPipe` クラスである。

**public List getFOAF(Node startNode, List linkPipes) メソッド**

第一引数で指定したノードオブジェクト(実際は、ユーザ識別オブジェクトとなる)を始点とし、相互の知人リンクを持つ 2 クリークの距離のユーザとのリンクを示すオブジェクト `core.logic.network.DConnectFOAFLinkPipe` (方向つきパイプオブジェクト) を格納したリストとして、`java.util.List` インターフェース型のオブジェクトとして返す。この `List` インターフェースの実装型は、`java.util.ArrayList` である。

なお、第二引数にて、始点ノードから 1 クリークの距離のノードとのリンクを示す `core.framework.network.Link` インターフェースの実装オブジェクトを指定する。

なお、戻り値の `core.logic.network.DConnectFOAFLinkPipe` オブジェクトは、

core.framework.network.Link インターフェースの実装クラスのインスタンスである。

表 5.15: core.logic.network.DConnectFOAFLinkPipe クラス

<<abstract class>> core.logic.network.DConnectFOAFLinkPipe	
<b>フィールド</b>	
- Node startNode	始点のノード
- Node endNode	終点のノード
- List hubNodes	中間のノード
<b>メソッド</b>	
+ Node getStarNode ()	始点のノードオブジェクトを取得
+ java.lang.String getStarNodeId()	始点のノード識別子を取得
+ Node getEndNode ()	終点のノードオブジェクトを取得
+ java.lang.String getEndNodeId()	終点のノード識別子を取得
+ Node getFOAFNode ()	getEndNode () の別名
+ java.lang.String getFOAFNodeId()	getEndNodeId() の別名
+ boolean hasDirection()	有向リンクであるかどうかを取得
+ boolean isActivated()	リンクが有効かどうかを取得
+ void setActivated(boolean)	ノードの有効フラグを設定
+ void setDirection(boolean)	ノードの有向フラグを設定
+ void setStartNode(Node)	始点ノードを設定
+ void setEndNode(Node)	終点ノードを設定
+ void setFOAFNode(Node)	setFOAFNode(Node) の別名
+ java.util.List getHubNodes()	中間のノードのリストを取得
+ void setHubNodes(java.util.List hubNodes)	中間のノードのリストを設定
+ void addHubNode(Node hubNode)	中間のノードをリストに追加

先ほど述べた core.logic.network.DConnectLinkPipe クラスが、1 クリークの距離を持つ 2 つのノード間のリンクを示すオブジェクトであったのに対し、core.logic.network.DConnectFOAFLinkPipe クラスは、2 クリークの距離を持つ 2 つのノード間のリンクを示すオブジェクトであり、2 つのノード間に存在する中間ノードの情報を保持する。実際には、この中間ノードは、2 ユーザの間の共通の友人 = 仲介者を示すユーザ識別オブジェクトのことである。

## ユーザ間のランデブー確立

ネットワーク演算の結果を受けて、ユーザ間でランデブーを確立する方法は、ネットワーク演算の結果として得られるリンクオブジェクトから、ユーザ識別オブジェクトを取得し、その getter メソッドにて、ランデブー確立のための各種必要なデータを得る。

```
List foafLinkPipes = engine.getFOAF(startNode, linkPipes);
for(int i=0; foafLinkPipes.size(); i++){
Node endNode = ((DConnectFOAFLinkPipe) foafLinkPipes).getEndNode();
sendMessage(endNode.getCom_id());
}
```

## 5.7 メッセージの送受信

本説では、本システムで扱われるメッセージの送受信について、一連の操作プロセス別に説明する。

### 5.7.1 人材検索ユーザからの人材検索メールの送信

人材検索ユーザから、人材検索メールが、知人や知人の知人に送信されるプロセスにおいては、メッセージ処理クラス `FPMailLogic` インターフェースの実装クラスのメソッドを用いる。デフォルトの実装クラスは、`core.logic.mail.FPWebMailLogic` クラスである。

ユーザは、人材検索メールを作成し、上記インターフェースのメソッドによって送信する。人材検索メールには、検索する人材に関する要件情報、自分自身に関する個人情報、クリーク毎のユーザに対する情報開示の制御のためのアクセス制御リストが含まれる。人材検索メールの作成画面を生成する、送信ボタンを提供するなど、これら、人材検索メールの作成を支援するのは、アプリケーション層の機能である。なお、送信されるメッセージのフォーマットについては後で述べる。

ユーザが作成した人材検索メールは、上記のメソッドによって、1 クリーク（ユーザの知人）および 2 クリークの距離（ユーザの知人の知人）のユーザに送信される。なお、本研究における実証実験においては  $N$  クリーク（但し、 $N < 2$  の整数）の距離に対するメッセージの送信は行わない。また、ユーザの設定次第では 1 クリークあるいは 2 クリークの距離のユーザにのみメッセージが送信されることもある。



表 5.16: core.framework.mail.FPMailLogic インターフェース

<<interface>> core.framework.mail.FPMailLogic	
<b>メソッド</b>	
+ void send1HopMail(Node startNode, String subject, String body, Link linkPipe, boolean queuing, String media_type)	1 クリークの距離の知人へのメッセージ送信
# StringBuffer get1HopMsgBody (String original_body, UserCert startUser)	1 クリークの距離の知人へのメッセージの本文を生成・取得
+ void send2HopMail(Node startNode, String subject, String body, Link linkPipe, boolean queuing, String media_type)	2 クリーク以上の距離の知人へのメッセージ送信
# StringBuffer get2HopMsgBody (String msg_body_org, DCUserCertBeans startUser, List hubNodes, String mediaType, String from_hash, String to_hash)	2 クリーク以上の距離の知人へのメッセージの本文を生成・取得
+ String getReplyURI(Node startNode, Node foafNode, int format)	返信識別 URI (後述) を取得する
<b>既知の実装クラスの一覧</b>	
core.logic.mail.FPWebMailLogic	

### 5.7.2 受信ユーザに提供される機能

前項で述べた、人材検索メールの送信プロセスが実行されると、前節までに述べた知人ネットワークの演算処理によって得た情報に従い、送信ユーザから 1 クリークないし 2 クリークの距離に存在するユーザに対して人材検索メールが配信される。

人材検索メールは送信ユーザが設定した情報開示の制御のためのアクセス制御リストに従い、メールの配信時に一部の情報が削除された形で、受信ユーザに配信される。

本項では、人材検索メールが配信されたユーザに対して提供される機能について説明する。送信者ユーザには主に以下の 2 つの機能が提供される。

#### 仲介者情報の提供

2 クリークの距離のユーザが人材検索メールを受信した場合は、必ず受信ユーザと送信ユーザの間に存在する共通の知人のリストが提供される。この機能は本人材マッチングモデル上最も重要な機能である。

表 5.17: 受信ユーザに提供される機能

機能名	説明
仲介者情報	送信ユーザと受信ユーザの間の距離が 2 クリークの場合、その間に存在する共通の知人リストが提供される機能
返信識別 URI	送信ユーザの識別子が開示されていない際に、送信ユーザへの返信を実現するため、一時的な返信用 URI を提供する機能

仲介者のリストは、メールの送信時に利用するメソッド

`core.logic.mail.FPWebMailLogic:: send2HopMail()` メソッドにおいて、知人ネットワークの演算クラスの `DCMatchingEngine:: getHubNodes(Node, Node)` メソッドを利用し、メッセージに追加される。

この機能により、共通の友人にアクセスすることで、受信ユーザは送信ユーザに対して返信をする前に、送信ユーザに対するより詳細な情報を得ることが出来る。また、受信ユーザは後述する方法によって送信ユーザに直接返信を行うことも可能であるが、さらに、共通の友人を紹介を依頼することも可能となる。

#### 返信識別 URI の提供

人材検索メールは先に述べたように、送信ユーザが設定した情報開示の制御のためのアクセス制御リストに従い、メールの配信時に一部の情報が削除された形で受信ユーザに配信される。その際、もっとも重要な情報が送信ユーザを一意に識別する情報である。通常、電子メールなどのメッセージでは、この情報は送信者情報として、From ヘッダーや Sender ヘッダーに掲載される送信者の電子メールアドレスとなる。しかし本システムでは、メールアドレス等のユーザを一意に識別可能な情報は、設定次第で開示しないことができる。一方で、人材マッチングを成立させる以上、受信ユーザが送信ユーザに対して何らかの手段によって到達可能でなければならない。そこで本システムでは、送信者の特定のために、返信識別 URI という仕組みを使う。

返信識別 URI は、送信ユーザを一意に識別することなく、受信ユーザがメッセージの送信ユーザに対して返信を行えるようにするもので、受信ユーザに配信されるメッセージに挿入される。この処理は、`core.framework.mail.FPMailLogic` インターフェースのメソッド `getReplyURI(Node startNode, Node foafNode)` によって取得した返信識別 URI を `core.logic.mail.FPWebMailLogic:: send2HopMail()` メソッドにおいて、受信ユーザに配信されるメッセージに挿入することで実現する。

返信識別 URI の書式は以下の通りである。

サーバ URL + J2EE アプリケーションコンテキスト名 + 返信用 Action にバインドされている URL + 文字列 "?" + 返信識別パラメータ

サーバ URL は、本システムを動作させている J2EE サーバ（ないし、Servlet コンテナが動作するサーバ）の識別子であり、J2EE 環境にて、

```
request.getScheme()+ "://" +request.getServerName()+ ":"
+ request.getServerPort()
```

（但し、request は、HttpServletRequest のインスタンス）

にて取得できる文字列である。J2EE アプリケーションコンテキスト名は、同じく J2EE 環境にて、

```
request.getContextPath()
```

（但し、request は、HttpServletRequest のインスタンス）

にて取得できる文字列である。

返信用 Action にバインドされている URL は、返信用 Action クラスの実装クラスを呼び出すために Struts の設定ファイル struts-config.xml に定義されている URL である。返信用 Action クラスの実装クラスは任意の名前がかまわないが、返信識別 URI を HTTP のリクエストパラメータから取得し、ユーザからの返信処理を行うクラスである

core.framework.mail.FPMailLogic インターフェースの実装クラスを呼び出す処理を行う必要がある。

返信識別パラメータは、以下の 2 通りがある。

表 5.18: 返信識別パラメータの書式

	書式
1	vta= (送信者を一時識別する文字列)
2	vua= (返信者を一時識別する文字列) &vta= (送信者を一時識別する文字列)

書式 1 は、送信者を一時的に識別するための文字列のみで構成される。一方、書式 2 は書式 1 に加え、返信者を一時的に識別する文字列が含まれている。書式 2 は、主にクライアントに携帯電話を用いたアプリケーションで利用することを想定している書式であり、書式 2 によって、返信用 Action へアクセスすると、自動的にユーザ認証される。書式 1 を用いた場合は、返信用 Action へアクセス時に、ログイン処理を要求される。

送信者を一時識別する文字列及び返信者を一時識別する文字列は、送信者或いは受信者のユーザ識別子、および、人材検索メールの送信日時に、システムにあらかじめ設定した

文字列を足し合わせたものに対して、MD5 アルゴリズムによってハッシュ値を求めたものである。

以下は、返信識別 URI の一例である。

```
https://j2ee.sfc.wide.ad.jp:80/dconnect/FPWebReplydo?vua=16b166
3b089c857b2006a3a2fda83373&vta=02a2f86c61c3253534a4867814f57e50
```

返信識別 URI を構成する各要素は、人材検索メール一時保持テーブル connect\_msg\_hashes に格納される。格納される時間は、送信ユーザが指定した期間か、システム管理者が JNDI[66] に登録した期間のいずれか短い方である。この期間の間であれば、受信ユーザ、送信ユーザ双方にとって、自分自身を一意に識別する情報 (E-Mail アドレス等) を開示しなくとも、本システム上では相互に到達可能となる。

表 5.19: 人材検索メール保持テーブル connect\_msg\_hashes のスキーマ

属性の説明	属性名	データ型	制限等
始点ユーザ識別子	from_id	可変長文字列	NOT NULL
終点ユーザ識別子	to_id	可変長文字列	NOT NULL
始点ユーザへの メッセージ返信用 識別子	com_addr	可変長文字列	NOT NULL
人材検索メール識 別子	auth_key	可変長文字列	NOT NULL
要求日時	create_timestamp	TIMESTAMP	デフォルト値::現在時刻
有効フラグ	available	BOOL	NOT NULL, デフォルト 値::true

主キー : from\_id, to\_id の複合キー

インデックス : from\_id, to\_id

この返信識別 URI は、本システムを利用して送受信したメッセージには常に追加されるため、最初の何度かのメッセージは、お互いが匿名 (ここでは、ユーザが本システム以外のコミュニティにおいて自分自身を特定可能な識別情報を開示しない状態を匿名と定義する) のまま行う、といったことも可能である。しかし、その際も、匿名でコミュニケーションを行う 2 者の間に存在する、共通の知人に関する情報は常に両者に提供される。

## 5.8 人材検索メールのメッセージフォーマット

本システムにおいてユーザ間で送受信される人材検索メールのメッセージフォーマットを説明する。人材検索メールには、検索する人材に関する要件情報、自分自身に関する個

人情報、クリーク毎のユーザに対する情報開示の制御のためのアクセス制御リストが含まれる。

人材検索メールは、上記の各要素をパートとして保持するマルチパートの MIME メッセージとして表現される。

各パートは、文字列或いは Java のオブジェクトとして構成される。

表 5.20: 人材検索メールのメッセージフォーマット

パートの内容	属性の型	補足
検索する人材に関する要件情報	テキスト	送信ユーザが記述する、検索した人材に関する要件情報
送信ユーザ自身に関する個人情報	UserCert 型の JavaBeans	送信ユーザ自身の個人情報を格納した UserCert 型の JavaBeans オブジェクト
クリーク毎のユーザに対する情報開示の制御のためのアクセス制御リスト	DCUserACL 型の JavaBeans	送信ユーザ自身に関する個人情報の UserCert 型の JavaBeans オブジェクトのプロパティのそれぞれに対し、クリーク毎のユーザに開示するかどうかの制御情報が格納された DCUserACL 型の JavaBeans オブジェクト

## 5.9 アプリケーション層

### 5.9.1 アプリケーション層の実装

アプリケーション層は、本システムを利用するコミュニティの特徴や達成する目的に応じて、実際にユーザにサービスを提供する機能や提供するクライアントの種類、インターフェースなどを柔軟に変更することを実現するために、コアモジュールとユーザとの間に存在する層である。

本研究では、第 5.1 節において述べたように、Web アプリケーション版の実装と、携帯電話版の実装を用意した。

アプリケーション層の重要な機能の一つとして、本システムを利用するコミュニティの特徴や達成する目的に応じて変化するユーザの個人情報項目の差異を、吸収することが挙げられる。

ユーザの個人情報は、5.4.1 にて述べたように、`core.framework.foundation.DCUserCert` インターフェースを実装するオブジェクトである。この制限さえ満たしていれば、アプリ

ケーションの実装者は、ユーザの個人情報の項目を自由に追加することが出来る。その際、`core.framework.foundation.DCUserCert` インターフェースの実装クラスにプロパティとして項目を追加した上で、データベース等のデータ永続化リソースから、データを取得しそのクラスのインスタンスを生成するため

`core.framework.foundation.UserUtils` の各メソッドを実装した実装クラスを作成する。

`core.framework.foundation.UserUtils` インターフェースは、ユーザの個人情報を管理するオブジェクトの作成、管理機能を定義したインターフェースであり、アプリケーションは、このインターフェースの実装クラスを利用し、ユーザの個人情報を管理する。

他のアプリケーション層の具体的な役割として、コアモジュールが提供する機能と、クライアントの種類、インターフェースとの間の疎結合化が挙げられる。これらの役割は、5.2 で述べたレイヤリング設計に従い厳格に定義されたファンクション（ロジック）層の機能の呼び出し手続き（*Degrees Connect* API）に従って、アプリケーションを実装することで、実現する。

### 5.9.2 Web 版実装

本研究で用意した実装の一つは、Web 版実装である。

Web 版実装は、共通の目的や興味を持ったユーザ間で利用されるコミュニティサイト等に埋め込まれる形で統合される J2EE 実装であり、継続的に存続するコミュニティにおける継続的な人材マッチングのサービスを提供する。

実装に用いた技術は、5.2 にて述べたとおり Java Servlet / JSP, Struts 等, J2EE を用いた。図 5.6 に、ページ構成を示す。

ユーザは、本実装の上で、知人リンクの追加・編集、人材検索メールの送受信等、*Degrees Connect* の基本機能をすべて利用できる。

図 5.7 に、Web 版実装のメイン画面（ログイン後トップページ）のインターフェースを示す。このインターフェースは、次章で述べる慶應義塾大学 SFC Incubation Village 研究コンソーシアムでの実験サイト SIV Business Networking の実際の画面である。

左側のリストがユーザの知人リストとなる。このリストを用いて知人リンクを管理する。本サイトの使い方が表示されている右側の部分は、代わりに、人材検索メッセージの作成画面を表示する、或いは、本実装を埋め込むサイトのコンテンツを代わりに表示することもできる。

### 5.9.3 携帯電話版実装

本研究で用意した実装の二つ目は、携帯電話版実装である。

本実装は、イベントなどの時限的でリアルタイム性が重要視される状況での使用を想定している。実装に用いた技術は、5.2 にて述べたとおり J2EE を用いた。

図 5.8 に、ページ構成を示す。特徴的な点は、携帯電話からのアクセスの利便性を考え、ログイン方法が 2 種類存在する点である。通常の Web フォームからユーザ識別子とユーザ認証用パスワードを入力して行う方法に加え、5.7.2 にて述べた返信識別 URI を用いてログインする方法を用意した。

図 5.9 に、携帯電話版実装のインターフェースを示す。このインターフェースは、次章で述べる国際学会 ISAGA2003 での実験用アプリケーションの実際の画面である。

左側の画面が人材検索メールの返信画面である。また、送信画面もほぼ同様のインターフェースである。右側の画面は、受信した人材検索メールを読んでいる画面で、メールの下部に、共通の知人リストが表示されている様子が分る。

本実装は、View テクノロジーに JSP1.2 をベースに Struts タグライブラリと JSTL を用いており、少々の変更で、ユーザインターフェースの表示言語を変更できる。図 5.9 は、国際学会で用いるために英語表示であるが、次章で述べる SIV Networking Seminar での実験では、日本語表示を行った。

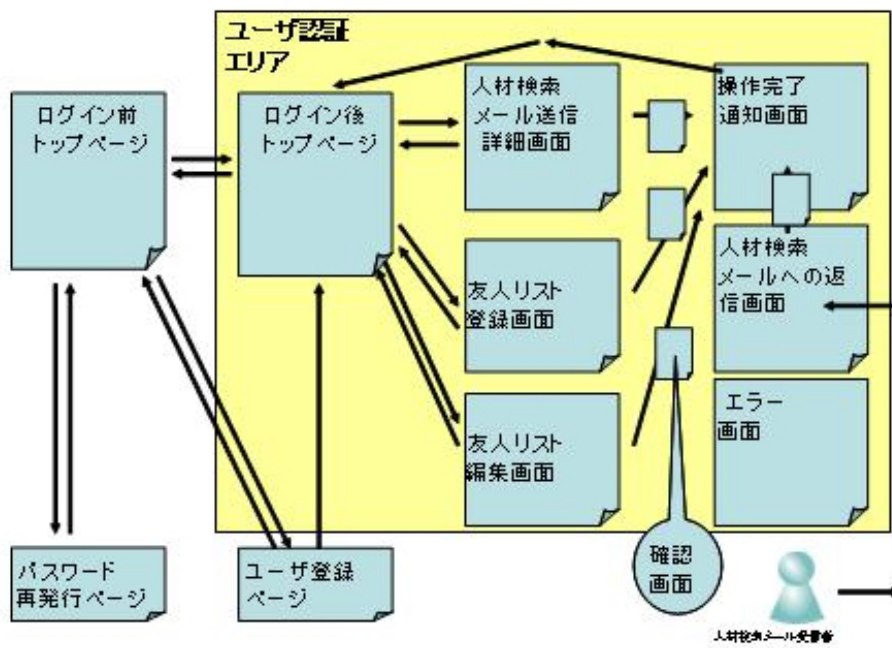


図 5.6: Web 版実装のページ構成



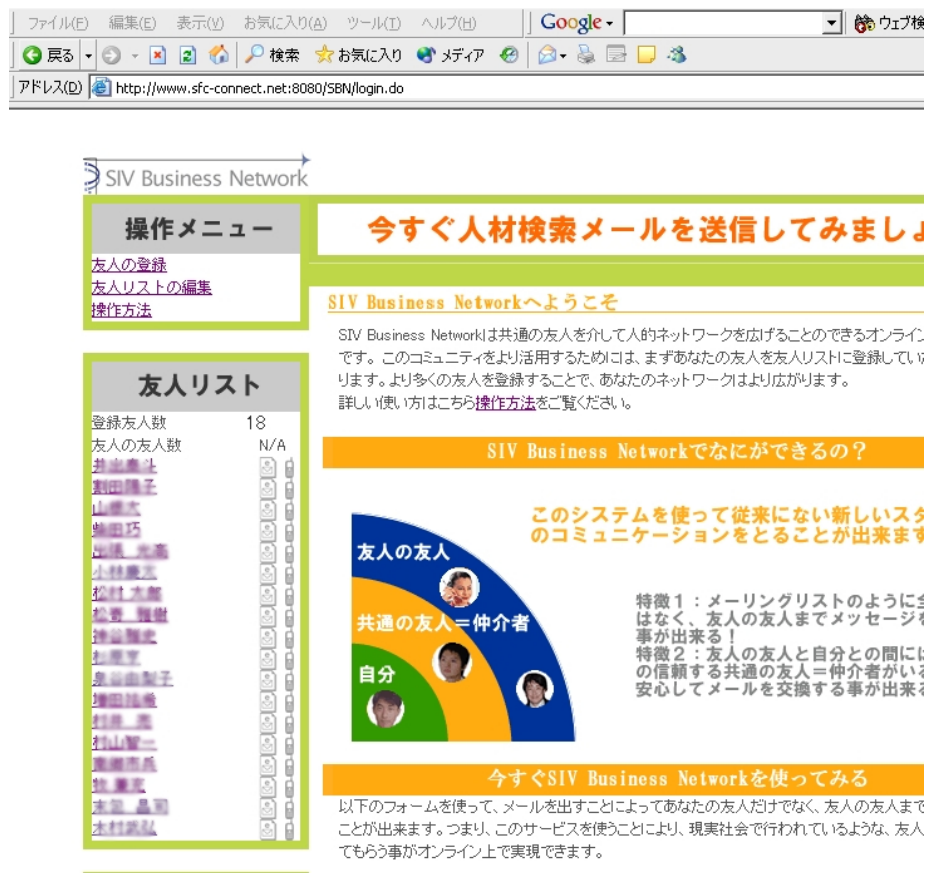


図 5.7: Web 版実装のインターフェース

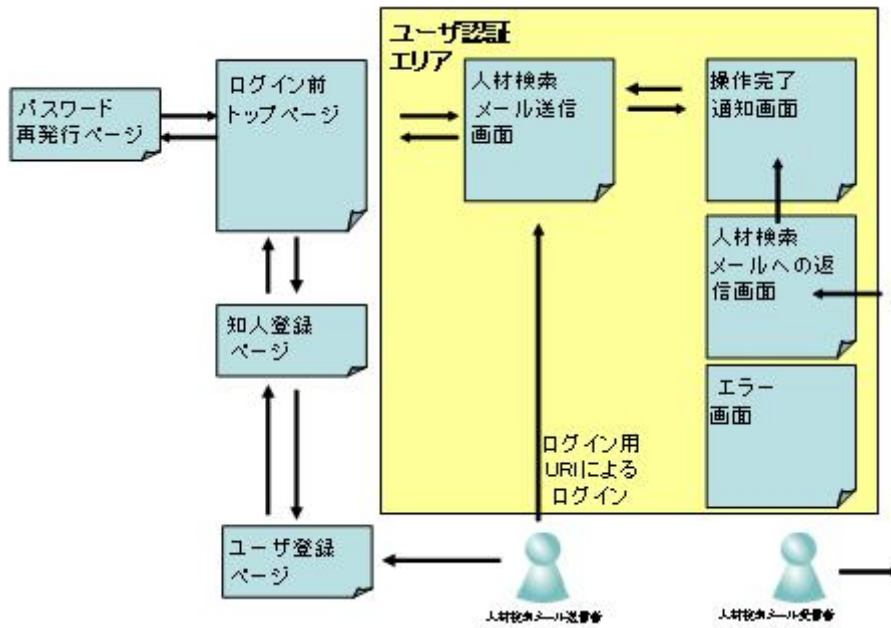


図 5.8: 携帯電話版実装のページ構成

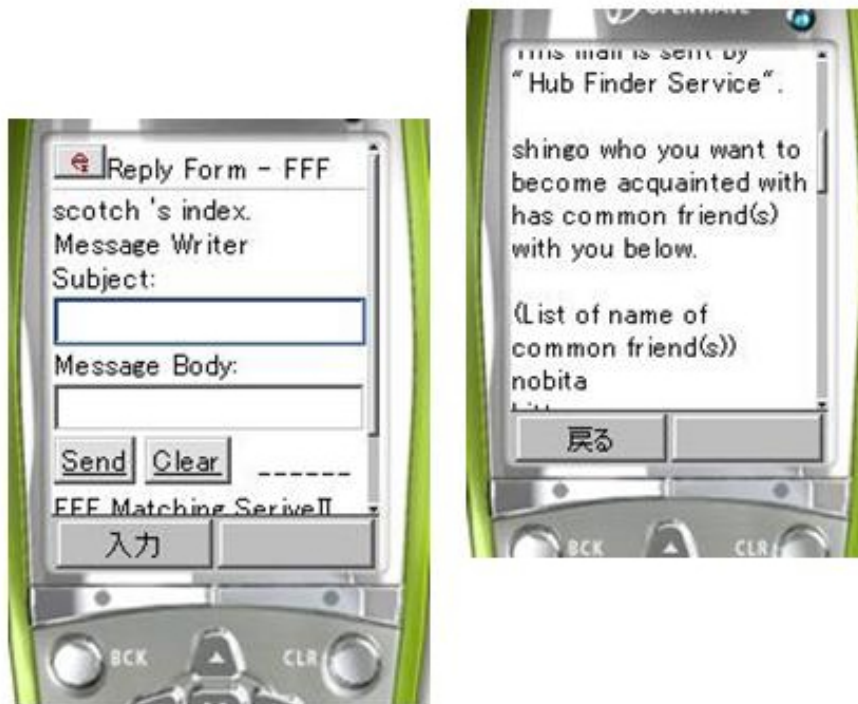


図 5.9: 携帯電話版実装のインターフェース

## 第6章 Degrees Connect の評価

### 6.1 要求定義の確認

本研究で提案するマッチングモデルの有効性と問題点、今後の課題を明らかに検証するため、人材検索を目的とする複数のコミュニティを対象に、実装したシステムを運用し実証実験を行った。

対象とするコミュニティは、同窓会や学会といった 3.1 で述べた性質を持つ複数のコミュニティである。

また検証項目は、3.3 で述べた点を検証するものであり、具体的には表 6.1 の項目を設定した。

表 6.1: 検証実験における検証項目

#### 実装の動作検証

- 目的を達成するための機能がきちんと実装され、コストやスケーラビリティの面で実運用に耐えうるか

#### モデル評価 1; 人材検索メールの配信範囲の妥当性

- 現在、2 クリークに固定している人材検索メールの配信範囲が妥当であるかどうか

#### モデル評価 2; 情報開示における自発性パラドクスの解消

- 本モデルが、知人ネットワークにおける信頼の結節点の存在を用いることで、情報開示のリスクを低減できていたかを検証

#### マッチ率への効果の定量評価

- 本モデルが 2 者の引き合わせ率 (マッチ率) に対して与える影響の定量評価

#### その他; モデルの利点、問題点の把握

- その他、モデルに関して、ユーザからの定性評価を行い、利点や問題点を把握

## 6.2 モデルの妥当性に関する先行実験

### 6.2.1 実施要領

本研究に先立ち、筆者自身による先行研究として以前、本研究のモデルの有効性を検証する実験を行った。その実験を先行実験として紹介する。先行実験は 2002 年 1 月に行った。実施要領は表 6.2 の通りである。

表 6.2: 先行実験の実施要領

目的	モデルの有効性（プライバシー管理に関して）
実験期間	2002/1
実験手法	アンケート調査
調査対象	慶応湘南藤沢キャンパスに所属し Web アプリケーションを使うユーザ 106 人
調査内容	Web アプリケーションでよく使われる個人情報項目計 31 について、それぞれ、従来モデルと、本モデルとの 2 つの異なる環境下において、公開するかしないかの定量比較

目的は本研究のモデルの有効性を検証するために、第 2 章において紹介した bolt をはじめとする従来モデル（ユーザの個人情報を集中管理し、他の全てのユーザにユーザの情報が開示されてしまうモデル）と、本研究のモデルのそれぞれにおいて、どれだけ個人情報を開示してもよいかをそれぞれ回答して貰った。個人情報は、Web アプリケーションでよく使われる個人情報項目を中心に計 31 項目である。具体的には、氏名、ハンドル名、E-Mail アドレス、誕生日、性別、住所、自宅電話番号、自宅 FAX 番号、携帯電話番号、携帯 Mail アドレス、Web の URL、インスタントメッセージの ID 郵便番号、住所等である。

### 6.2.2 調査結果

有効回答数は、96 回答であり、回答の内訳は、男性 65%、女性 35%であった。また、学年分布も学部 1 年 25%、学部 2 年 21%、学部 3 年 19%、学部 4 年 23%、その他 11%と平均的な分布であった。

回答の結果、本モデルでは承諾した割合は、従来モデルでの承諾数より平均して 37%大きくなった。また、住所、自宅電話番号、自宅 FAX 番号、携帯電話番号、携帯 Mail アドレスに関しては、従来モデルでの開示承諾率が 5%に満たないのに対し、本モデルでは平均で約 50%に向上した。

なおこの実験の詳細は、「自己情報コントロール権を実現した人材マッチングシステムの研究」[67]を参照いただきたい。

### 6.2.3 先行研究の結論

この実験の結果から、本モデルは従来モデルと比較して、情報開示に対するリスクが低減し、情報開示の承諾率が高くなることが示された。

## 6.3 実証実験の目的と実施概要

本節では、本研究で提案するマッチングモデルの有効性と問題点、今後の課題を明らかに検証するために行ったいくつかの実証実験について述べる。

### 6.3.1 実証実験 1 (ISAGA2003) の実施概要

先行実験を踏まえ、第 4 章、第 5 章で説明したシステムを用いて、全ての検証項目を対象に実証実験 1 を行った。実証実験 1 の実施要領は表 6.3 の通りである。

表 6.3: 実証実験 1 (ISAGA) の実施要領

目的	モデルの有効性、システムの有効性の検証
実験期間	2003/8/25-2003/8/30 第 34 回国際シミュレーション&ゲーミング学会
実験手法	本システムの携帯版実装のデモンストレーション
調査対象	第 34 回国際シミュレーション&ゲーミング学会参加者 47 人
調査内容	実際に本システムを用いて、人材検索を行ってもらった
主な検証項目	・実装の動作検証・モデル評価・マッチ率への効果の定量評価・モデルの利点や問題点に関する定性評価

第 34 回国際シミュレーション&ゲーミング学会(以下、ISAGA2003 と記する)とは、国際シミュレーション&ゲーミング学会大会 (ISAGA) が毎年開催する会議であり、年間のシミュレーションおよびゲーミング研究の成果を共有するとともに、研究者・教育実践家・実務家の世界的交流を目的とするものである。2003 年は、8 月 25 日(月)から 8 月 29 日(金)までの 5 日間、千葉県木更津市のかずさアカデミアパークを会場として開催された。

世界 35 か国から約 430 名の参加者が参加し、様々な学問領域の研究者による、デモンストレーション、発表、ポスターセッションなどが行われた。この学会では、発表者がデモンストレーションという形で、参加者に自身の研究成果を体験、或いは、実験に参加してもらうということが行われ、それらのデモンストレーションは、5 日間行われた学会に通して参加した参加者(日本人 100 人、外国人 100 人程度)を対象に行われた。本実験もそのような中、THE BOX Project のデモンストレーション企画として行われた実験 [68] の一部である。

### 6.3.2 実証実験 1(ISAGA2003) における調査方法

ISAGA2003 で行った実験は、湘南藤沢キャンパスの村井純研究室と加藤文俊研究室のメンバーによる合同研究プロジェクト THE BOX Project (筆者もメンバーとして参加) のデモンストレーションの一部とした。THE BOX Project は、先ほども述べたように、筆者自身も参加するプロジェクトで、RFID タグを用いた新しいコミュニケーション分析の研究手法を確立することを目的とした研究プロジェクトである。ISAGA2003 では、メンバーシップが限定され、時間軸的にも数日で終了するといった性質を持つ学会発表というイベントにおいて、RFID タグを用いることで、学会のコミュニケーションを分析、促進するためのデモンストレーションを行った。

そのデモンストレーションでは、先ほど説明した本システムの携帯電話版実装を用いて、RFID タグを利用した人材マッチングサービスを提供した。具体的には以下のサービスを提供した。

1. 「特に新規にお話をしたい人」と、お話をする機会の創出をサポートするサービス
2. 研究のコラボレータとの出会い

前者のサービスは、「特に新規にお話をしたい人」と、お話をする機会の創出をサポートするサービスである。

後者のサービスは、自分の知人の知人に、自分の研究に興味を持つ人がいるかどうかを検索できるサービスである。

両方のサービスとも、実験に参加していただくことを了解していただいた方に、参加時に ISAGA2003 参加者の知人を学会の参加者名簿からリストアップしていただいた。また、ISAGA2003 への参加を通し、新規に知り合いになりたい参加者を名簿からリストアップしていただいた。

その情報から ISAGA2003 参加者間の知人ネットワークを構築し、前者のサービスにおいては、知人ネットワークの情報から、参加者と参加者が登録した「新規に知り合いになりたい参加者」との間に存在する共通の友人を携帯メールで案内した。また後者においては、参加者の携帯電話から本システムにアクセスすることで、参加者の知人と、参加者の知人の知人に対して、自身の研究内容、ISAGA2003 で行うセッションの告知などを行えるサービスを提供した。

また、参加者に RFID タグを配布し、RFID による参加者のリアルタイムな位置情報を、人材マッチングの際に、知人の範囲で提供した。

また、携帯電話を所有していない参加者へは、携帯電話を貸し出した。

### 6.3.3 実証実験 1(ISAGA2003) の実施結果

実証実験 1(ISAGA2003) への参加人数は 33 人であった。うち、10 人はレンタル携帯電話による参加であり、参加する時間や利用していただく機能に制限があった。

また、国際学会でのデモンストレーションという性質もあり、参加していただいた人の割合がそれほど高くなかったため、双方向の知人関係がなかなか構築度されなかった。したがって、実際には、ユーザ間が知人関係であっても、どちらかがデモンストレーションに参加なさっていない場合は、知人ネットワークが形成されなかった。また、ISAGA2003 では会期中からの参加される方が多く、参加人数の増加がゆるやかであったため、その点も双方向の知人関係の構築を妨げた。また、1 日参加者が大半であったため人材マッチングにも影響を与えた。

このように、本モデルの実験環境として十分な環境ではなかったため、特にマッチ率の定量評価においては精度の高い結果が期待できなかったが、いくつかの検証項目を検証することが出来た。

より詳細な実施結果等は THE BOX Project のプロジェクト報告ページ [69] を参考いただきたい。

### 6.3.4 実証実験 2(HCD) の実施概要

実証実験 2 は、検証項目の一つである情報の配信範囲の妥当性の検証のために、2003 年 10 月に行った実証実験である。

その実施要領は表 6.4 の通りである。

表 6.4: 実証実験 2(HCD) の実施要領

目的	モデルの有効性（配信範囲の妥当性の検証）
実験期間	2003/10/11（慶應義塾大学湘南藤沢キャンパス、ホームカミングディ 2003）
実験手法	Web アプリケーションのログ解析
調査対象	慶応湘南藤沢キャンパスの卒業生 135 人
調査内容	参加者間の知人ネットワークを取得し、知人ネットワークを考察
主な検証項目	・モデル評価（情報の配信範囲の妥当性）・知人ネットワークのフリースケール度の検証

慶應義塾大学湘南藤沢キャンパスは、1990 年に新設された慶應義塾大学のキャンパスの一つである。学際指向の高い特徴的な理念とカリキュラムを持つキャンパスである。湘南藤沢キャンパスでは 2002 年度より、年に 1 度、卒業生がキャンパスに戻り、旧友や恩師と

再会するイベントが行われている。このイベントがホームカミングディ(略称 HCD)である。本実験は、2 回目の開催となるホームカミングディ 2003 の一コンテンツ「イマココ」のデータを分析したものである。

イマココは、湘南藤沢キャンパスの村井純研究室と加藤文俊研究室のメンバーによる共同研究プロジェクト THE BOX Project (筆者もメンバーとして参加) が企画、実行したサービスで、RF-ID タグと携帯電話を用いた Alumni 再会促進サービスである。

「イマココ」では、参加していただくユーザ(卒業生)の方々に、ユーザ登録をお願いした。参加者は、ユーザ登録時から、現役時代に所属していた研究会、サークル、アドバイザーグループを登録していただいた。研究会とは、一般の大学で言うゼミや研究室にあたるものであり、アドバイザーグループとはクラスに代わるグループである。

### 6.3.5 実証実験 3(SIV Networking Seminar) の実施概要

実証実験 1 を経て、同様の検証項目の検証を目的として、対象とするコミュニティを変更して、実証実験 3 を行った。

実証実験 3 の対象コミュニティは、慶應義塾大学 SFC Incubation Village 研究コンソーシアム [70] (以下、SIV と記する) が主催する交流会 SIV Networking Seminar である。SIV とは、慶應義塾大学 湘南藤沢キャンパスをベースにビジネスインキュベーションプラットフォームのモデル研究を目的とする研究コンソーシアムである。その中で SIV Networking Seminar (以下、SNS と記する) は、慶應義塾大学湘南藤沢キャンパスの現役生、卒業生を中心とした、起業家と起業家支援者が集まる交流イベントである。

SNS は、数ヶ月に 1 回の頻度で開催される。1 回あたり 3 時間弱程度の長さで行われる交流イベントで、1 時間程度の講演会と、2 時間の立食パーティー形式での交流会が行われる。

したがって、同一組織への所属、目的の一致性といった点において、本モデルの活用ターゲットとして理想環境に近く、交流会では、起業家と起業家支援者とのマッチングが生じるため、本モデルを運用するには最適な対象であることから、実験対象として選定した。

実証実験 3 の詳細は表 6.5 の通りである。

### 6.3.6 実証実験 3(SIV Networking Seminar) における調査方法

SNS における実験では、ISAGA2003 での実験同様、携帯電話版実装を用いて人材マッチングサービスを提供した。具体的には以下のサービスを提供した。

1. 起業家と支援者との出会い



表 6.5: 実証実験 3(SIV Networking Seminar) の実施概要

目的	モデルの有効性、システムの有効性の検証
実験期間	2003/10/31 第 3 回 SIV Networking Seminar
実験手法	本システムの携帯版実装のデモンストレーション
調査対象	第 3 回 SIV Networking Seminar 参加者 100 人
調査内容	実際に本システムを用いて、人材検索を行ってもらった
主な検証項目	・実装の動作検証・モデル評価・マッチ率への効果の定量評価・モデルの利点や問題点に関する定性評価

このサービスは、ISAGA2003 での 2 つ目のサービスである「研究のコラボレータとの出会い」と同様のサービスである。自分の知人の知人に、自分の活動に興味を持つ人がいるかどうかを検索できるサービスである。つまり、起業家にとっては自身のビジネスを支援してもらおう支援者を検索でき、支援者にとっては自身の支援を必要とする起業家を検索できる。

### 6.3.7 実証実験 3(SIV Networking Seminar) の実施結果

今回は、SNS の参加時に、実験へ参加していただけるかをお聞きした。また、SNS の参加申込は SNS 開催の約 1 週間前に締め切られる点や、SNS はすでに数回開催されており過去の参加者名簿が存在した点で、知人ネットワークの構築に関して ISAGA2003 のときの問題を回避することができた。

SNS への参加人数は 100 人、うち本実験への参加を承諾していただいたのは 42 人である。

先ほど述べたように、実験参加の方法及び知人の登録の方法が ISAGA2003 と異なったため、ISAGA2003 と比較して、より正確に実際のユーザ間の知人関係が取得できたと考えられる。実際に、双方向の知人リンクの構築度が高くなった。

一方で、講演会と立食パーティという性質から、携帯電話での操作を行う時間が限られ、ユーザ間でメッセージをやり取りする頻度が少なかった。

### 6.3.8 実証実験 4(SIV Business Networking) の実施概要

前述の実験を経て、同じ検証項目の検証を目的とし、アプリケーションの種類を変更して、実証実験 4 を行った。

実証実験 4 の対象コミュニティは、実証実験 3 と同じ、慶應義塾大学 SFC Incubation Village 研究コンソーシアム（以下、SIV と記する）が主催する交流会 SIV Networking Seminar の参加者である。

実証実験 4 では、Web アプリケーション実装を用いて行う。SNS に参加した方々へ、人材マッチングサービスを SNS の前後にも日常的に受けられるサービスを提供するコミュニティサイトを構築する。

実証実験 4 の詳細は以下の通りである。

表 6.6: 実証実験 4(SIV Business Networking) の実施概要

目的	モデルの有効性、システムの有効性の検証
実験期間	2003/12/18-
実験手法	本システムの Web 版実装の運用
調査対象	第 3 回 SIV Networking Seminar 参加者 100 人
調査内容	実際に本システムを用いて、人材検索を行ってもらった
主な検証項目	・実装の動作検証・モデル評価・マッチ率への効果の定量評価・モデルの利点や問題点に関する定性評価

本実験 3 は、実験と比較してスパンの長い継続的な実験である。したがって、より知人ネットワークの構築に時間をかけることが可能であるため、時間の経過に従い、より実際の知人関係に近い知人ネットワークができることが期待できる。また、人材マッチングを行う時間も長くなるため、スパンの短い実験では明らかにならなかったケースや問題点などが明らかにされると考える。

### 6.3.9 実証実験 4(SIV Business Networking) の調査方法

本システムの Web アプリケーション実装を用いて、SNS の参加者を対象にしたコミュニティサイト SIV Business Networking (以下、SBN と記す) を実装した。

SNS に参加した方々に、SBN への参加案内を出し、SBN サイト上で人材マッチングシステムを利用していただいた。

SBN は、第 5 章で説明したように、Web サイト上で、知人の検索、登録と、人材検索メールの配信が出来る。ユーザは Web サイトで SBN の参加者リストの一覧を閲覧でき、そのリストから知人を登録する。登録をすることで、登録相手にも本サービスの告知が届く。このような仕組みのため、比較的短い時間で、実際の知人関係が登録されると考える。

ユーザには、SBN サイト上で、知人を登録し、人材検索メールを送って人材を検索していただいた。

## 6.4 実験結果と検証項目の考察

### 6.4.1 実装の動作検証

#### 実装の動作検証の概要

実装の動作検証は、実証実験 1,3,4 にて行った。

実装の動作検証に関しては、Degrees Connect のコアモジュール及び携帯電話版実装が、設計どおりに実装され、本研究で想定するターゲットにおいて、実用可能な安定性、コストパフォーマンスが得られるかを検証した。

実証実験 1(ISAGA2003) および実証実験 3(SIV Networking Seminar) では、学会発表や立食パーティといったリアルタイム性が高いシチュエーションでの人材マッチングが行われた。このようなシチュエーションでは、ユーザ間で短い時間に多くのメッセージの交換が行われることが予測される。また、クライアントに携帯電話を用い、携帯電話へのメッセージを配信する。携帯電話へのメッセージ配信においては、携帯電話サービスを提供する通信会社のネットワークを経由する。通信会社のネットワークは、SPAMメールの防止のため、同一のメールアドレスから、短い時間に多くのメッセージが送信されると、通信が切断されてしまい、数日間メッセージが送信できなくなってしまう場合がある。第 5 章で述べた Degrees Connect プロトタイプ実装では、知人ネットワーク演算における経路計算アルゴリズムによって、同一のユーザに複数のメッセージを送信しない仕組みが実装されている。また、携帯電話実装においては、同一アドレスから短い時間に、多くのメッセージを同時に送信しないように、メッセージをキューに入れ、送信するタイミングを調整する機能を実装している。

#### 実装の動作検証の結果

実証実験 1(ISAGA2003) における動作検証では、会期中最も負荷がかかった瞬間 50 クエリ/分、100 通/分のメール送信に対し、携帯電話の通信会社の SPAM フィルタにかけられることなく安定動作した。

実証実験 3(SIV Networking Seminar) における動作検証においても、携帯電話の通信会社の SPAM フィルタにかけられることなく安定動作した。また、インターフェースの国際化対応機能が正常に動作し、日本語環境においても不具合なく機能した。

#### 6.4.2 モデル評価 1; 人材検索メールの配信範囲の妥当性に関する検証

##### 情報配信の範囲に関する考察

知人関係を用いた情報配信を考える際に、その範囲の妥当性を検証する必要がある。つまり、発信する情報は、いくつの知人間を越えるのが最適か、という点である。ネットワーク理論の世界ではクリークという単位で示す。本研究では、1 クリークは、お互いが知人関係にある 2 者間の距離を指すこととなる。つまり、あるユーザから、そのユーザの知人までの距離が 1 クリークとなる。

ネットワーク理論の最新の研究によると、知人ネットワークはスケールフリーネットワークであることが分かってきた [71]。スケールフリーネットワークは、あるノードが所有するリンクの数がランダムではなく「べき法則」に従ったネットワークであり、少数のノードが多数のリンクを持つことで、ネットワークのハブのような機能を果たしている。したがって、同じクリークの距離であるユーザから到達できるリンクの数はランダムネットワークより一般的に多くなる。このことは、より少ないクリークで、多くのノードに到達する可能性が高くなることを示している。また、ハブ機能を果たすノードは、多くのノードとリンクを持っていることを意味するため、多くのノードにとってハブの存在は身近である。したがって、多くのノードは容易にハブを経由し、多数のノードに少ないクリーク数にて到達できてしまうため、ランダムネットワークと比較して、クリーク数の変化に対する到達可能なノードの数の変化は緩やかであると仮定できる。

一方、3.3.2 で述べた本モデルの有効性は、情報配信範囲が 2 クリークである場合のみ成り立つ。したがって、情報配信範囲を 3 クリーク以上にしなければ、アプリケーションとして十分な利用目的を達成できないという状況で無い限り、配信範囲を 2 クリークに設定することが妥当な選択であると考えられる。

##### 情報配信の範囲に関する検証概要

6.4.2 での考察による仮定の検証と、情報の配信範囲の妥当性の検証のために、検証実験 1,2,4 でのデータを元に分析を行った。検証実験 1 が学会発表コミュニティでの実験、検証実験 2,4 が同窓会の性質を持つコミュニティである。なお、検証実験 4 は、検証実験 3 のユーザを含むため、検証実験 3 での分析は行っていない。

検証実験 1 (ISAGA2003)、検証実験 4 (SIV Business Networking) では、ユーザ間によって登録された実際の知人関係を用いて、任意のユーザ (ノード) からの  $N$  クリークによる到達可能なノード数を分析した。

検証実験 2 (HCD) では、ホームカミングディという企画の中で行わせていただいた実験であったため、扱えるデータに制限があった。本来であれば、参加者全員に知人を申告し

てもらうのが最適であるが、人数の規模からして現実的な方法ではなかったため、6.3.4 で述べたように、現役時代に所属していた研究会、サークル、アドバイザリーグループというデータを利用した。具体的には、研究会、サークル、アドバイザリーグループを擬人化し、所属していた人と擬人化された研究会、サークル、アドバイザリーグループの間に知人リンクを形成し、知人ネットワークを可視化して、その全体像を把握、考察した。

### 検証結果

検証実験 1(ISAGA) と実証実験 4(SIV Business Networking) でのデータにおける分析結果を示す。図 6.1 は、各ユーザから 1 クリック、2 クリック、3 クリックで到達可能なノード数の累計の平均値をグラフ化したものである。SBN と書かれているグラフが実証実験 4(SIV Business Networking) の結果であり、AVG と書かれているグラフは、検証実験 1(ISAGA) と実証実験 4(SIV Business Networking) の平均値である。

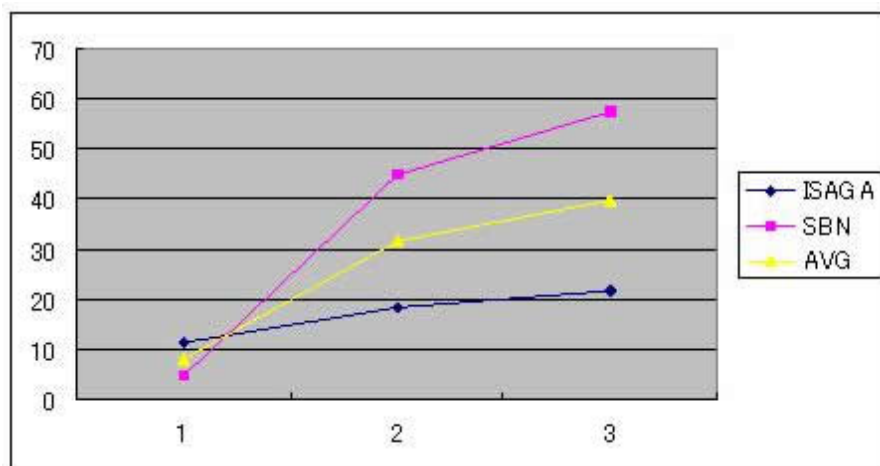


図 6.1: クリック数による到達可能なノード数

また、図 6.2 は、検証実験 1(ISAGA) と実証実験 4(SIV Business Networking) において、全ユーザに対する各クリック数で到達できるユーザの数の累計比である。

図 6.1、図 6.2 の結果、1 クリックから 2 クリックへの変化と比較し 2 クリックから 3 クリックへの変化は緩やかであることが分かった。したがって、2 クリックに制限することで、仲介者の存在というメリットを優先することの妥当性が裏付けられた。

なお、検証実験 1(ISAGA) では、学会発表というコミュニティの性質からか、実証実験 4(SIV Business Networking) と比較し、既に多くの知人関係が構築されており、全ユーザに対する到達可能なユーザ数の累計比は大きい。この現象は、実験への参加告知の方法の差が要因であると考えられる。実証実験 4(SIV Business Networking) では、SIV Networking

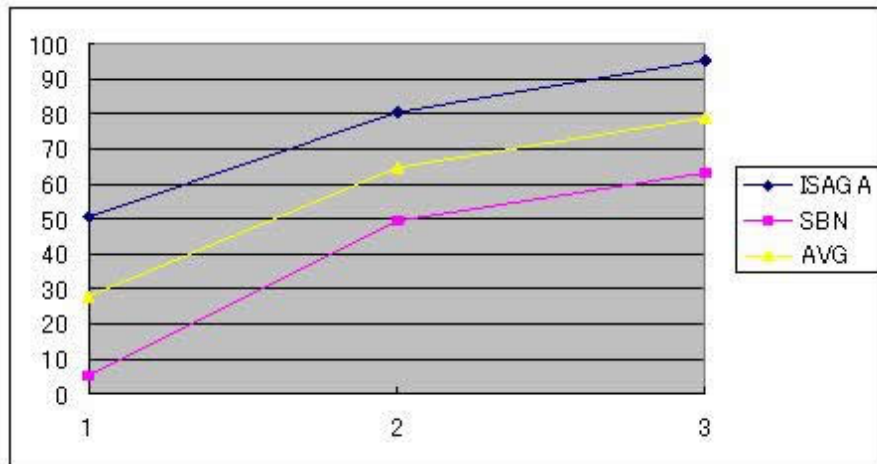


図 6.2: 全ユーザに対する到達可能なユーザ数の累計比

Seminar の参加者全員に対し、実験への参加告知を行えたが、検証実験 1(ISAGA) では学会会場にて参加を呼びかけたため、実験に参加していただいたユーザは知人同士で誘い合う形で参加したケースが多いと考えられる。

次に、検証実験 2(HCD) での検証結果であるが、可視化された知人ネットワーク図については、大きさが大きいため一部を 6.3 に示した。完全な図は、THE BOX Project のサイト [69] を参照いただきたい。ホームカミングディ 2003 企画自体への参加者数が、予定の 2000 人を大幅に下回ったため、イマココ企画への参加者数も、当初の予想の 1000 人を大きく下回り、135 人に止まった。よって、サンプル数の不足により、精度が十分とはいいがたいが、ユースケースに近い環境で、実際のユーザに関する情報が得られた点では非常に成果がある。

このデータにおいては、一部のユーザが多くの組織とのリンクを持っているといった特徴が確認された。また、その関係が異なる専門領域を超えて形成されていることがわかった(図 6.3 参照)。そのことは、人材マッチングにとって重要な要素である。

図 6.4 は、実証実験 4(SIV Business Networking) におけるユーザ毎の知人リンク数である。検証実験 2(HCD) の検証結果において、一部のユーザが多くの組織とリンクを持っていたということが、実証実験 4(SIV Business Networking) においても示された。このことは、知人ネットワークはスケールフリーネットワークであることを示し、6.4.2 における考察を裏付けるものである。

#### 検証結果の考察

6.4.2 にて述べたように、すべての検証結果において、6.4.2 における考察の妥当性が検

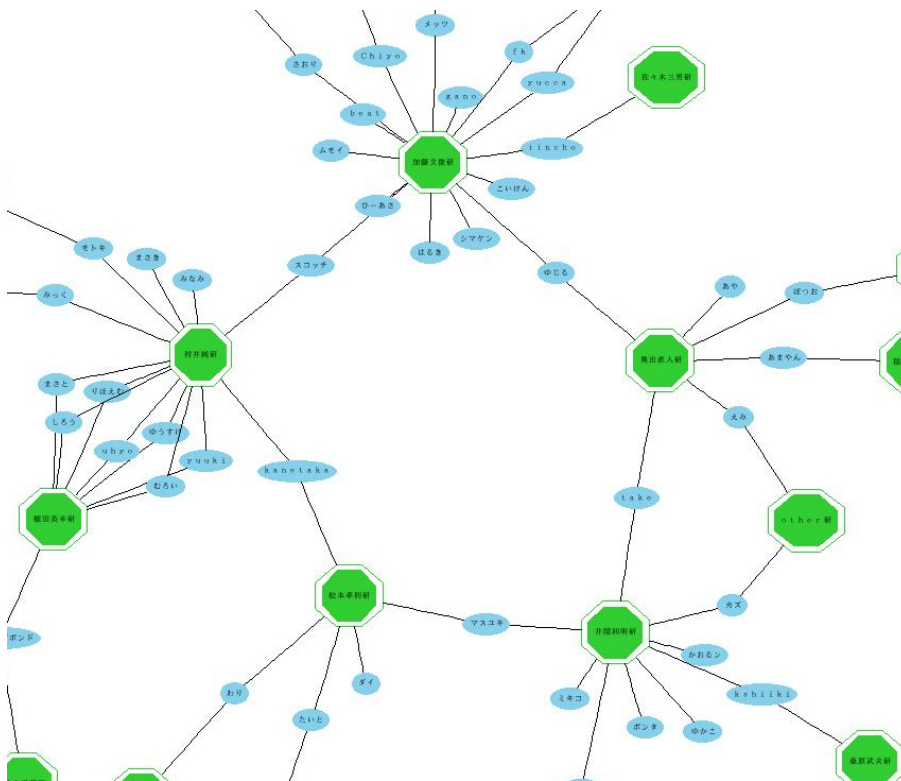


図 6.3: ホームカミングディ2003 の知人ネットワーク

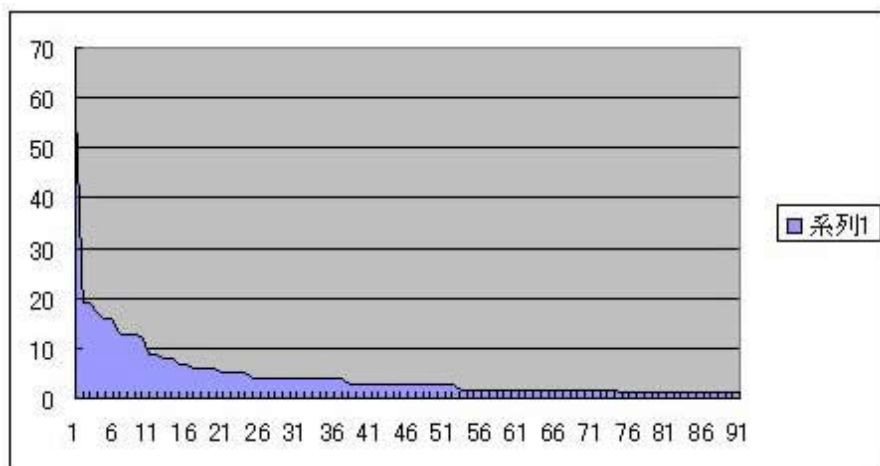


図 6.4: ユーザ毎の知人リンク数

証された。

また、ホームカミングディ2003 で図示された知人ネットワークの図の結果、本モデルは、ユーザの属性を越えた多様な属性を持つユーザ間でのマッチングを可能にすることが示唆された。

### 6.4.3 モデル評価 2; 情報開示における自発性パラドクスの解消

#### 検証方法

情報開示における自発性パラドクスの解消に対する本モデルの有効性を評価するため、以下の 2 点を検証する。

1. 情報開示において、知人関係を用いることにより、ユーザに対する自主選択可能な選択肢の提供が実現できたか
2. 仲介者の存在が、コミュニケーション相手の選定や、信頼性の高いコミュニケーションを実現できたか

#### 自主選択可能な選択肢の提供に関する検証

本検証項目は、検証実験 1,3,4 にて行った。最初に、1 つ目の自主選択可能な選択肢に関して考察する。本モデルは、ユーザ間の知人関係によって構成される知人ネットワークにおける信頼の結節点の存在を用いることで、人材を検索するユーザに対して、自己の情報の開示範囲と内容を細かく設定することを可能とする。また、人材検索メールを返信するユーザにとっては仲介者の存在によって、従来のマッチングモデルと比べ、受信する情報に対して比較的高い信頼性を得ることができる。また、ユーザ自身の情報を自身の自主的な選択によって保護しながら、最終的に自身の情報を公開するタイミングを自己決定できる。

これらの点は、手段である以下の機能が、正しく動作しユーザに提供されていることで、実現される。

1. 知人ネットワークによって情報の開示範囲の 2 クリックへの制限
2. 知人ネットワークによるクリック毎の情報の開示内容の制限
3. 共通の仲介者を通した 1 対 1 のランデブーにおける、任意のタイミングでの情報開示

第一の機能である「情報の開示範囲の 2 クリックへの制限」は、5.6 に述べたネットワーク演算モジュールによって実現され、すべての実装においてユーザに提供された。

また、第二の機能である「クリック毎の情報の開示内容の制限」に関しては、すべての実装において、デフォルト値である「2 クリック以上の距離へユーザの識別情報の非公開」



が実現された。ユーザに対し、2 クリーク以上の距離のユーザに対し識別情報の公開/非公開の自主選択を可能とした。

また、実証実験 1 においては識別情報に加え、RFID タグの利用によって取得したユーザの位置情報も同様に、2 クリーク以上の距離のユーザへは非公開にした。このことはアンケートにて、75%の被験者が情報開示のリスク低下の効果があったとしている。

第三の機能は、人材を検索する検索ユーザからの要件に対して、2 クリーク離れたユーザが、検索ユーザとの共通の知人である仲介者を介して、1 対 1 のランデブーを確立した後に、任意のタイミングでの情報開示を行えるようにするものである。この機能は、5.7.2 にて述べた返信識別 URI の機能を利用することで、前述の第一の機能による識別情報の公開/非公開の自主選択を、要件に返信をするユーザにも提供することで実現した。このことにより、要件に返信をするユーザは、識別情報を開示せずに、検索ユーザとコミュニケーションを取ることも可能となり、共通の仲介者を通した匿名コミュニケーションによる任意のタイミングでの情報開示を可能とした。

このように、知人ネットワークのモデル化を通して、ユーザに情報開示に関する自主選択可能な選択肢を提供し、情報開示リスクの自己制御を可能とした。

#### 仲介者の存在に関する検証

次に、仲介者の存在がマッチングにて引き合わされた 2 者間において、コミュニケーション相手の選定や、信頼性の高いコミュニケーションの実現に対して与えた影響を、利点と問題点の両方から考察する。

本システムによって引き合わされた 2 者間のコミュニケーションは、常に 2 者間の共通の知人である仲介者のリストが提供される。ユーザは、この仲介者の存在がもたらす機能を利用することで、従来のマッチングモデルでは検証が困難であったユーザのコンピテンシーの客観的正確性の検証を行うことができる。また、お互いが識別情報を明かさずにコミュニケーションをとっている場合であっても、仲介者の存在により、従来の匿名コミュニケーションと比較して、信頼性の高いコミュニケーションが実現できる。

これら仲介者の存在が実現する効果は、以下の機能の実現によって成り立つ。

1. 2 クリークのユーザ間のコミュニケーションにおける共通の仲介者の表示
2. 仲介者を用いたコミュニケーション相手の詳細に関する情報取得手段

これらの機能は、本研究で用いた 2 つの実装において実現され、ユーザに提供された。実際にそれらの機能がユーザに与えた効果は、以下の通りである。

1. コミュニケーション内容の信頼性が高まる (実験 1; 日本人大学生)

2. コミュニケーション相手をより親密に感じることができる (実験 1; 日本人大学生)
3. 突然共通の友人が判明するというのは、驚き半分嬉しさ半分で、その後にとっても役に立つと思う (実験 1; 日本人大学生)
4. I think that the use of a common friend system certainly is a good way to operate a network organisation. (実験 1; 欧州大学生)
5. I think that meeting people through two hops might be a bit too much. The strength of the concept lies in the usage of common friends. (実験 1; 欧州大学生)
6. 共通の知人の名前や所属から、コミュニケーション相手がどういった人物かある程度予測できる (実験 3,4; 社会人支援者)
7. 共通の知人から、コミュニケーション相手の素性がある程度察することができる (実験 3,4; 社会人支援者)
8. 共通の知人の存在によって、親しみがわく (実験 3,4; 社会人支援者)
9. 誰の友人かが分かることで安心感がある (実験 3,4; 社会人起業家)
10. 「灯台下暗し」的な人間関係が合った場合、共通の知人リストは有効だ。例えば、自分と相手との間で共通の知人に対する情報交換などができ、そのことを通して、お互いに意外なスキル等が発見できる。(実験 3,4; 学生起業志望者)

検証結果によって、仲介者がもたらす効果を実現するために必要なモデル上の重要な機能がユーザに提供され、ヒアリングの結果、仲介者がもたらす効果が確認された。

#### 6.4.4 マッチ率への効果の定量評価

マッチ率への効果の定量評価では、本システムを用いたマッチングにおけるマッチ率に関して定量評価する。従来のマッチングモデルと比較して、本システムのマッチ率が大きく低下せず、実用可能性に影響を与えないことを検証する。

##### マッチ率への効果の測定方法

マッチ率への効果の定量評価は、実証実験 1 において、従来モデルと本モデルの間での定量比較を行った。

実証実験 3, 実証実験 4 においては、システムのログを用い、要件の内容によるコンタクト率、マッチ率等の分析を行った。

## 6.4.5 マッチ率への効果の検証結果と考察

実証実験 1 では、従来モデルの人材マッチングシステムと本システムでの人材マッチングの間で、マッチ率の定量比較を試みたが、6.3.3 にて述べたように、対象を 2 つに分けた厳密な比較実験はできなかった。したがって、学会後にユーザへアンケートを実施し、本システムを通したマッチのうち、もし本システムを用いなければ出会わなかった人の数を回答してもらった。アンケートの回答数は 12 である。表 6.7 は、その回答とサーバログから得たデータを表にまとめたものである。

表 6.7: 実証実験 1 のデータ

データ項目	数値
人材検索メールの送信回数	2.17 通/ユーザ
人材検索メール知人からの受信数	2.57 通/ユーザ
人材検索メール知人の知人からの受信数	2.71 通/ユーザ
コンタクト数	1.86 人/ユーザ
知人同士を紹介した数 (アンケートより)	1.14 人/ユーザ
本システムによってのみ成立したマッチ数 (アンケートより)	0.71 人/ユーザ

被験者が 1 人あたり平均して約 1.86 人以上のコンタクトを成立させた。また、本システムを用いなければ出会わなかったと想定される人数は、被験者が 1 人あたり平均 0.71 人になる。

この値はあくまで、被験者一人一人の主観に基づく想定である。実験に参加していただき、かつ、アンケートにも答えていただいた方は本モデルのコンセプトに肯定的であり、そのことが値を大きくする方向に影響を与えた可能性がある。また、実験コミュニティが学会であり、立食パーティ等の交流イベントが存在する社交的な場において、本システムを用いなくとも出会っていた人数を多めに想定し、この値が小さくなった可能性も考えられる。しかしながら、これらのバイアスを考慮したとしても、本モデルが 3.1 にて想定したコミュニティのひとつである学会において人材マッチングを行う上で、この値は実用可能な範囲の値であるといえる。

検証実験 3 (SIV Networking Seminar) においては、要件の内容によるコンタクト率、マッチ率等の分析において、興味深い結果が現れた。

やり取りされた 8 通のメッセージの内、コンタクトがあったのは、4 通であった。コンタクトが成立した 4 通と、成立しなかった 4 通の間で、興味深い差があった。検索の内容によって、コンタクト率が上下したのである。

コンタクトが成立したメッセージは、人材そのものを規定してしまうような要件ではなく、人材を持っている情報などを検索する要件の方が、コンタクトの成立率が高い。

コンタクト率が高い要件の例：

携帯電話ビジネスについて詳しい人をご存知の方はいらっしゃいますか

コンタクト率が低い要件の例：

優秀な Java 技術者を探しています

この点は、ユーザからのヒアリングにおいても指摘されている。(6.4.6 参照)

表 6.8 は、サーバログから得たデータ、及び、ユーザからのアンケート結果をまとめたものである。サーバログからのデータはアクティブユーザー人当たりの平均である。アクティブユーザは、SIV Business Networking に 2 回以上ログインしたユーザと定義する。また、その他はアンケート回答者一人当たりの平均値である。(アンケートの回答数は 10)

SIV Business Networking は、SNS に参加した方々に、参加案内を出すという形でユーザアカウントを発行した。したがって、サイトのユーザアカウント数と、実際に利用しているアクティブユーザの間に差があるため、実際に利用しているユーザとして、2 回以上ログインした 22 ユーザをアクティブユーザとして定義した。

知人ネットワークを用いる本モデルの特徴上、知人関係にあるユーザ間のコミュニケーションのうち、本システムを用いずに行われているコミュニケーションはサーバログには記録されない。したがって、実際にはさらに多くのメッセージが交換されていると予測される。

したがって、アンケートの回答項目の値の方が、サーバログの値よりも大きくなる傾向がある。さらに、アンケートへ回答していただいたユーザは標準的なユーザと比較して、より頻繁に本システムを利用している可能性が高い。

表 6.8: 実証実験 4 のデータ

データ項目	数値
人材検索メールの送信回数	1.14 通/アクティブユーザ
人材検索メールの受信数	2.43 通/アクティブユーザ
コンタクト数	1.00 人/アクティブユーザ
マッチ率 (アンケートより)	0.86 人/回答ユーザ
受信メールへの興味をもった数 (アンケートより)	0.86 通/回答ユーザ
受信メールへの返信数 (アンケートより)	0.43 通/回答ユーザ
知人同士の紹介数 (アンケートより)	0.57 通/回答ユーザ

#### 6.4.6 その他; モデルの利点、問題点の把握

本節では、ユーザへのヒアリングによって明らかになった、前述の検証項目以外のモデルの利点、問題点について述べる。

##### ユーザへのヒアリングによる定性評価

本モデルに対する被験者の反応は非常に大きな反応があった。

利用した感想に関しては、以下のようなコメントを得た。

- 友達との出会い方をこうやってプロデュースできるのだなと感心した。もっと活用されれば、システムだけでなく、そのコミュニティ全体が相当活性化されると思う。(実験 1; 日本人大学生)
- 出会うことが出来た人を見ると、日常の出会いではあまり出会えない人と出会えたと感じる。(実験 1; 日本人大学生)
- 自分が持っているネットワークを超えて、なかなか知り合うことのできない人と知り合えるおもしろさがある(実験 4; 社会人起業家)
- 実際に友人ネットワークを使人探しをするには、同じメールを何通も送らねばならずそれなりにめんどくさい。そういう意味で、このシステムは便利だ(実験 4; 社会人支援者)
- 仕事の上での仲間さがしなどに使いたい。(実験 4; 社会人支援者)
- 新規アイデアの感想を求めたりプロモーションをかけるときに利用したい。(実験 4; 社会人支援者)
- 誰に聞いたらいいかわからない事を聞きたい時や、仕事関係で協力者を探したい時に利用できる(実験 4; 社会人支援者)
- 人材を検索する時、きっと誰かはいそうだけど、具体的に誰、という具体的な名前が挙がらないときにも使える(実験 4; 社会人支援者)
- 使ってみて友人の友人の方からよきアドバイスを頂いたので非常に役に立っている(実験 4; 学生起業準備中)
- たまにはスポーツの呼びかけも良いかも(実験 4; 学生起業準備中)

また、2 クリークはなれた人への返信に関しては、

- メールを送信するというのは抵抗がある
- どの程度具体的な情報を返信してよいか分らない
- 他の人がどのような利用をしているかが分らないので、利用を戸惑う
- 年代が違う方（社会人の方）からメールが届いた時は、少し返信に気を使い、心理的抵抗はあった
- 自分のスキルが相手の要求を満たすかが不安で返信を戸惑う

といった意見の一方で、

- 仕組みを理解した上で参加しているので特に抵抗は感じない
- 誰の友人であるかが確認できるので、特に抵抗はない
- 友人の友人からのメールに返信する場合に、匿名（アドレスも非公開）にできる機能が役に立つ
- 自分がスキルの役に立てることであれば、特に抵抗は無い
- さまざまな人と接したいという積極的な目的があるため

という肯定的な意見も得られた。これらのことから、考察した点を以下に述べる。

#### 概念の理解と共通認識の重要性

知人ネットワークを活用したネットワーク上のコミュニケーションツールは現時点で一般的でないため、その概念をユーザに理解してもらうことが、返信率といった利用の促進において重要だと考えられる。また、現時点では、他のユーザがどのような使い方をしていいるか、どのような効果があったか、といった利用スタイルや具体的な活用方法に関する共通のノウハウが無い。また、知人関係を用いて探していた人材が見つかった場合、どのような人にお礼をするべきか、といったことに関する認識が形成されるにはまだ時間がかかると考えられる。ユーザ間でこれらの知識や認識が共有され、共通認識として形成されることが重要である。この点に対する今後の課題としては、システムにユーザ間の知識共有を促進する仕組みを取り入れることや、知人ネットワークを活用した人材マッチングモデルそのものを普及させ、より多くのユーザに利用してもらうことが挙げられる。

#### 活用コミュニティの性質

3.1 にて述べたとおり、ユーザ間の目的共有度の高さ、目的志向性の高さが、返信等システムの活用に当たっての障壁に関係していることが裏付けられた。

## 要件の内容

6.4.5 でも指摘したとおり、要件の難易度が高く、スペシフィックな要求は、受信ユーザー側の謙遜を生み、返信率への障害となる。高度な要求になればなるほど、或いは、スペシフィックな要求になればなるほど、求める人材が見つかりにくいことは、人材マッチングにおいて一般的に起こることである。ここで求める人材が見つかる確率に関して、その確率は、前述したコミュニティの目的共有度の高さ、目的志向性の高さ等によっても変化すると考えられる。また、前述した利用スタイルや具体的な活用方法に関する共通のノウハウが蓄積されることは、要件の難易度と人材が見つかる確率の間での最適点を、人材を検索するユーザが自分自身で判断することに容易にすると考えられる。

### 6.4.7 ケース; 仲介者の声

最後に、実証実験 4 におけるケースを紹介し、その際の仲介者へヒアリング結果を紹介、考察する。

本ケースは、SNS に参加した社会人支援者（営利企業勤務の会社員）による人材検索である。要件は、Zope という Web 技術のスキルを持つ技術者アルバイトの募集であり、そのようなニーズが社内にあったということである。人材検索ユーザは、要件に見合うユーザが

「直接知り合いにはいなかったが知り合いの知り合いにはいそうな気がした」

ということで、SBN を利用した人材検索に至った。

「契約には至っていませんが 2 通の返事があり、社内ニーズに応えることができました。」

という感想を頂いている。

この内 1 通の返信の仲介者にお話を頂いた。

この仲介者は、大学教員であり SNS を主催するユーザである。したがって、SBN のユーザコミュニティにおいて中心的な役割を担うユーザで、登録知人数も 58 人と最も多い。

よって、このユーザは、普段から知人同士を紹介することが多いとのことであるが、本システムによって、紹介を半自動化することができ、従来に比べて楽になったと感じているとコメントした。さらに今回のケースは、自身が仲介者として紹介をすることになった 2 者のことを、従来からお互い良く知っていたが、本ケースの要件に関して 2 者のニーズが一致する点を、仲介者ユーザ自身気づかず、本システムによって初めて気づいたとのことである。以下に、そのインタビューの要点を示す。

「SIV Networking Seminar の主催者という立場ですので、このコミュニティのハブとしての役割を担っています。当然多くの方に、紹介の仲介を頼まれる訳ですが、このシステムが導入されたことによって、その手間が大幅に下がりました。私自身お名前を知っていてもその方が持っている技能を全て把握している訳ではないので、そのために今までマッチングできなかった方同士が、本システムによりマッチングされる、という事例も発生しています。マッチングにおける「チャンスの向上」と「効率性」という二つの側面において本システムは有効であると思います。」

このケースは、6.4.6 にて紹介した実証実験 1 における日本人大学生のコメントの予期せぬ人同士のマッチングと同様であり、仲介者の立場からも予期せぬ人同士がマッチングされることがあるということを示している。

また、本モデルは、登録されている知人リンク数が多ければそれだけ仲介者に負担がかかると考えられる。しかし、知人リンクを多く持つユーザは、実社会上においても多くの知人関係を持っていることを意味するため、本モデルの存在に関わらず、日常的に知人同士の紹介を行っている可能性が高い。よって注目すべき点は、紹介行為における、本モデルによる負担と、従来の負担の間での比較である。その点において、本ケースでは、本モデルの負担面での有効性が示された。



## 第7章 結論

### 7.1 本研究の達成事項

#### 7.1.1 人材マッチングにおける自発性パラドクスを解消

実社会上における資源であるユーザ間の知人関係をコミュニケーションツールにモデル化すること、及び、そのモデル化においてユーザに対し情報開示における自主選択可能なオプションを提供することを通し、情報における自発性パラドクスを解消するマッチングモデルを実現した。

その実現は、次項から述べるいくつかの達成事項によってもたらされたものである。

#### 7.1.2 エンドユーザ主体の情報管理の実現

ユーザ間の知人関係によって構成される知人ネットワークにおける信頼の結節点の存在を用いることで、情報開示に関して、人材検索ユーザに対しては個人情報の開示リスクとマッチングの効用との間でのトレードオフ上に自主選択可能な選択肢を新たに提供した。この選択肢によって、検索ユーザの情報開示度を最適化し、送信者の自己情報コントロール権を強化した。また受信ユーザにとっても、自身のセンシティブ情報を自身の自主選択によって保護しながら、最終的に自身のセンシティブ情報を開示するか否かを決定できる。また、公開するタイミングも受信ユーザ自身による自己決定を可能とした。

#### 7.1.3 共通の知人（仲介者）が生み出す信頼性の高い人材マッチング

本モデルは、ユーザ間の知人関係というシステム外の要素の存在を所与のものとし、ユーザ間の知人関係（知人ネットワーク）が生み出す信頼性によって、情報配信の範囲の制限およびコミュニケーション相手の選定を行う。

実際にはユーザ間の知人関係（知人ネットワーク）及び、それが生み出す信頼構造には様々な種類があるが、本研究の実証実験において、本システムが想定しているユースケースのいくつかにおいては、2クリークという配信範囲の限定化と、そのことにより成立する共通の知人（仲介者）の存在が生み出す信頼の有効性を検証することが出来た。送信ユーザと受信ユーザの間に存在する共通の知人（仲介者）は、従来のマッチングモデルでは検

証が困難であったユーザのコンピテンシー等の客観的正確性が検証を可能とし、より信頼性のある人材マッチングを実現した。

#### 7.1.4 人材マッチングの効果について

マッチングの効果（マッチ率）に関して本モデルを用いることによるマッチ率の向上を定量的に実証することは出来なかったが、複数の検証実験を通して、本モデルが人材の引き合わせに妨げとなることはなく、本モデルによってのみ生まれた引き合わせ（仲介者にとっても予期せぬ引き合わせ）も実現した。したがって、本モデルは、マッチングの効果を低下させることなく、他の達成事項を実現できたと言える。

#### 7.1.5 高い実現性（スケーラビリティ、コストパフォーマンス）

本モデルは3.3.3にて述べたとおり、特定の役割やリソースに依存せず実現した。実運用による検証は行っていないが、ユーザ数の増加に対して高いコストパフォーマンスとスケーラビリティを持つモデルが実現できたといえる。

また、本研究が提案したモデルは、数々の実証実験が明らかにしたように、社会システムを大幅に変えることなく導入が可能であり、今後も様々なコミュニティを対象に運用を続けたいと考えている。

## 7.2 未達成及び不確定事項と今後の課題

### 7.2.1 モデル化における単純化と有効性の範囲について

本研究の立場は、特定の条件を備えたコミュニティにおいて人材マッチングというユースケースを限定した上で、実現可能性を踏まえたその有効性を検証するものである。その結果、本システムが想定しているユースケースのいくつかにおいては、2クリークという配信範囲の限定化と、そのことにより成立する共通の知人（仲介者）の存在が生み出す信頼の有効性を検証することが出来た。

このモデル化における単純化は、本モデルの実現に関するコストやスケーラビリティ、ユーザのモデルの理解の容易さといった点を生み出したが、一方で本モデルを適用するコミュニティ、ユースケースをさらに広げてゆく場合、このモデルの単純化がユースケースを限定するものとなる可能性がある。また、実際にはユーザ間の知人関係（知人ネットワーク）及び、それが生み出す信頼構造には様々な種類があるが、多くの性質のコミュニティにおいて、本モデルの単純化がどこまで有効であるかといった点を検証する必要がある。

これらの課題には、より多くの性質の異なるコミュニティでの検証を通し、コミュニティ

の性質と、マッチングへの有効性、有効なユースケースの範囲（応用範囲）の間の関係性について計測してゆく必要がある。

また、現在は2クリックといった到達距離のみがパラメータであるが、実社会においてはより多くのパラメータを利用している。例えば、ユースケースや目的に関わらず、すべての知人に対して人材検索メールを送るという現在のモデルが、人材を検索したいユーザにとって適切かどうか、議論の余地があろう。例えば、人材マッチングの要件によっては、人材検索メールを送る知人を限定したいこともあるであろう。

また、現在はユーザ間の知人関係は、存在するか、存在しないか、の2段階しかない。しかし、実社会においてはその間には無数の段階がある。そして、それらは時間の経過によって変化してゆくものである。単純化が生み出すメリットと人材マッチングの有効性、或いは応用範囲の拡大との間でのバランスを考えつつ、モデルの修正を検討する必要がある。

これらの点におけるモデルの修正において、検討可能な機能の一例としては以下のものが挙げられる。

#### リンクの優先順位付けおよび経路制御

知人ネットワークのリンクをより柔軟に管理する機構である。たとえば、ユーザの目的に応じて、一時的に一部のリンクを無効化する、優先順位をつける、といった機能や、時間経過と共に優先順位が変化するという機構である。

#### ユーザのニーズに応じた経路制御

ユーザの目的に応じて、どのユーザを経由して、或いは、経由しないで、メッセージを送受信するかどうか選択できる機能である。

#### 通信される情報の内容に応じた経路制御

ユーザ間でやり取りされるメッセージの内容に従って、最適な経路が自動的に決定される機能である。

### 7.2.2 実社会の事象のモデル化の精度について

本モデルの一番の特徴は、情報配信の範囲の制限およびコミュニケーション相手の選定における信頼判断をユーザの知人（共通の友人＝仲介者）に委譲する点である。したがって、仲介者がこの役割をどれだけ果たせるか、果たしてくれるか、に従って本モデルの有効性は変わってくる。したがって、第3章で議論したように、仲介者の働きが強化されや

すいコミュニティ、具体的には、ユーザ間に高度な信頼関係が築かれ、互恵性の高いコミュニティ、もしくは、構成員の間で共通の目的を共有し、目的志向性が高いコミュニティで本モデルは特に有効に働くと述べた。

したがって、本モデルの有効性を強化するためには、仲介者の働きの活性化する必要がある。そのためには、本システムを用いるコミュニティを選択する、仲介者へのインセンティブを与える、といったシステム外における要素を考慮に入れて考える必要がある。

本モデルが特に有効に働くコミュニティは、その要素としていかなる特徴を持つのかをより検証し、システムに反映することで、本モデルの有効性の強化や、適切なコミュニティ或いは応用範囲の選択に関する判断が促進される。

コミュニティの要素として考えられるのは、仲介者としてのコストが、コミュニティの構成員の間において、最適に分配されていることである。これらは、そのコストを負担に対して、ユーザが得られる利益を含めたインセンティブや評価、賞賛のスキームが最適化されていることと言える。

また、実証実験におけるケースで、マッチ率の高い要件と低い要件の差を考察した際、その差にはユーザの謙遜といった要因が少なからず働いていると推測された。ユーザ間の互恵性を高める意味でも謙遜といった要因を取り除く必要がある。その点において、返信ユーザへのインセンティブや評価、賞賛のスキームも必要である。

### 7.2.3 自主選択可能な選択肢に関して

本研究にて提案するモデルの有効性の一つは、知人ネットワークの活用によって、ユーザが情報開示を自主選択できる選択肢を提供する点である。この選択肢は、従来のマッチングモデルと比較して、ユーザの目的達成に対する効率性と情報開示に関するリスクとの間で、より最適な選択肢を提供するものであるが、ユーザへの利便性の面から、モデルの設計者がモデル化の段階において、ある程度ユーザにとって便利な選択肢のセットを設計し、提供することになる。

この際、モデル化を行った設計者（本研究では筆者）が設計した選択肢セットが、ユーザの目的達成にとって相応しいかどうかの検証が必要である。具体的には選択肢セットの妥当性によって、ユーザが信頼できると感じる度合いを低めていないか、という点である。

### 7.2.4 実装の改善

#### セキュリティとスケーラビリティ

実装の改善としては、システムの分散化が挙げられる。現在の実装は、主に実験の容易化を理由として一台のサーバに全てのデータを格納し、一つのCPUで全ての処理をして

いる。しかし、設計レベルでは peer to peer モデルになっており、ユーザ毎にデータを格納したクライアントをローカルに持つ実装も可能である。そうすることにより、データの分散化と計算資源の分散化が図れ、セキュリティとスケーラビリティが向上する。

### 7.3 今後に向けて

インターネットの登場は、それ以前に不可能であった多くのことを実現した。しかし、インターネットの本質である自律分散協調型コミュニケーションは、最もプリミティブな人間のコミュニケーションモデルであることも一方で事実である。近代、コミュニケーションを取り巻く技術の進化が、社会の進化のスピードに対し追いついていなかったために、コミュニケーションの形は他律集中型に変化せざるを得なかった。そのような中、デジタルテクノロジーの発展により、コミュニケーションの形が、再び本来の形に戻ってきたと筆者は考える。

つまり、インターネットの登場のショック、すなわちそれは、「あまりにシンプルなモデルによって我々人間の本質的なコミュニケーションを体現してしまったこと」によって、我々は以前に増して、より本質的な課題である「我々のコミュニケーションとは何か」といった疑問、或いは、我々の存在は何か、といった問いを直視する結果になったと言える。我々は、インターネットから、自己認識の再考、或いは、我々を取り巻く社会の仕組みを直視させられる結果になったのである。

本研究は、実社会上における資源であるユーザ間の知人関係をコミュニケーションツールにモデル化することで、情報における自発性パラドクスを解消するマッチングモデルを実現するものであるが、この試みは前述した課題の逆方向の行為であるといえる。しかしながら、本研究においても前述した課題は筆者に突きつけられた。我々が生きる実社会の仕組みを単純化し、コミュニケーションモデル化するといった本研究の行為は、そのプロセスにおいて、コミュニケーションモデルが逆に、筆者に対して、実社会の仕組みの本質を示唆するという現象が起きたのである。

このプロセスを実行する上で筆者が体験した、モデル化と本質との間での相互反復は、パラダイムシフトと言われる変化において、我々に提示された課題の克服手法、知識の創造手法の一つであると考える。

本研究は、今後も、「実社会の仕組みのモデル化 システム運用 システム運用から導き出される実社会の本質 実社会への働きかけ 「実社会」からのフィードバック 導き出された本質の再度のモデル化」といった相互反復を繰り返し、知識を創造してゆきたい。

## 謝辞

本研究を進めるにあたり、ご指導を頂いた村井純教授、金子郁容教授、國領二郎教授、加藤文俊助教授に感謝いたします。思えば今日の筆者がここに存在するのは、高校時代に金子郁容教授にネットワーキングへ招待されたからであります。村井純教授に教わり筆者の価値観に大きな影響を与えた自律分散協調の哲学、本研究はその哲学を筆者なりに体現したものです。このような機会をお与えいただいたことに心から感謝いたします。

また、執筆にあたって絶えずご指導とご助言を頂きました慶應義塾大学看護医療学部専任講師の宮川祥子氏、実験の計画や実施において常に援助をいただいた慶應義塾大学環境情報学部専任講師南政樹氏にも、大変お世話になりました。ここに感謝の意を表します。

また研究内容の議論において絶えずご指導と助言を頂いた村井研究室ならびに maui プロジェクトのみなさん、多くの議論を共にしたネットワークコミュニティプロジェクトのみんな、みなさんのおかげで、より深い視点で研究を進めることができました。またお忙しい中、キャンパスまで駆け付け助言を頂いた、株式会社東芝の若山史郎氏、慶應義塾大学理工学研究科の江木啓訓氏に改めて深く感謝いたします。また、慶應義塾大学政策・メディア研究科博士課程の石田剛朗氏、同課程の渡邊大輔氏に感謝いたします。

また修論にて忙しい時期に、共に研究室運営をした RG コーディネータのみなさん、ご迷惑をおかけしました。またその間、研究グループの運営を先導してくれた仲山昌宏氏に感謝いたします。来年度こそは我がグループの飛躍の年にしましょう。また、本波友行氏には、公私様々なシチュエーションでお世話になりました。来年からも社会人として、より一層の活躍を祈っています。

秘書の福永美夏さんを始め國領研究室の皆様にもお世話になりました。意外なアイセックつながりには、本研究の意義を示唆させるような大きな驚きを感じました。また、追い込みの時期に様々な点でお世話になった杉原亨氏をはじめ大学院棟で時間を共にしたすべてのみなさんにお礼を申し上げたいと思います。

数々の実証実験を始め、多くの活動を共にした THE BOX Project のメンバー、特にリーダー横山治己氏ならびに村井研究室の谷隆三郎氏には大変お世話になりました。共に活動ができてとても楽しかった。また、加藤文俊研究プロジェクトのみんな、特に中野友香氏、北野絢子氏、ISAGA2003 ヘススタッフとして参加したみんなに助けられ、本研究は完成しま

した。どうもありがとう。

また、実験にご協力いただいた、ISAGA2003 参加者のみなさん、徳田研究室のみなさん、慶応 SFC ホームカミングディ2003 実行委員のみなさん、ならびに卒業生の皆様、牧兼充氏をはじめとする SIV 事務局のみなさん、SIV Networking Seminar および SIV BusinessNetworking にご参加いただいた全ての皆様に、この場をお借りして、お礼を申し上げたいと思います。誠にありがとうございました。また、村井亮氏をはじめとする株式会社 Beat Communication の全てのスタッフにも大変お世話になりました。

そして最後に。本研究のすべてのきっかけは、sfc-connect プロジェクトにあります。今は共に異なった道を歩いている sfc-connect のオリジナルメンバー、第2期メンバーに感謝したい。そして、今後も、あの頃共に描いた学際的協調の推進と自律的なキャリア形成の支援というビジョンを一步一步形にしてゆくことをここに誓います。

この他、名前を挙げていくと切りがありませんが、自称学際主義者の筆者のある種変則的なコミュニティへ関わり方や、興味のまま次々に物事を始めてしまう性格に、時に振り回されながら寛大な心でそれを受け止め、温かい心でこれまで支えてくれた、すべての方々に感謝いたします。

以上をもって、謝辞といたします。

## 参考文献

- [1] WIDE プロジェクト. Wide プロジェクト報告書 2003. Technical report, WIDE プロジェクト, 2003.
- [2] 村井 純. インターネット. 岩波新書. 岩波書店, 1995.
- [3] Kiesler Sara and Sproull Lee. *Connections New Ways of Working in the Networked Organization*. MIT Press, 1991.
- [4] 國領 二郎. オープン・アーキテクチャ戦略 ネットワーク時代の協働モデル. ダイヤモンド社, 1999.
- [5] ダイヤモンド・ハーバード・ビジネス・レビュー編集部. ダイヤモンド・ハーバード・ビジネス・レビュー 2003年2月号特集: プロジェクト・マネジメント. ダイヤモンド, 3 2002.
- [6] Pink H. Daniel. *Free Agent Nation: The Future of Working for Yourself*. Warner Books, 2002.
- [7] Raymond S. Eric. *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary (O'Reilly Linux)*. O'Reilly & Associates, 1999.
- [8] 金子 郁容. ボランティア もう一つの情報社会. 岩波書店. 岩波書店, 1992.
- [9] Garrett Hardin. The tragedy of the commons. *Science*, 162:1243–1248, 1968.
- [10] 株式会社リクルート. リクルートナビ. <http://www.rikunabi.com/>.
- [11] 株式会社日経人材情報. 日経就職ナビ. <http://job.nikkei.co.jp/>.
- [12] 宮川 祥子. インターネット上の情報検索における社会的文脈の利用手法に関する研究 . PhD thesis, 慶應義塾大学, 2001.
- [13] R. Housley, W. Ford, W. Polk, and D. Solo. *Internet X.509 Public Key Infrastructure Certificate and CRL Profile*, January 1999. RFC 2459.
- [14] Garfinkel. *PGP: Pretty Good Privacy*. O'Reilly, 1996.



- [15] Mark Granovetter. The strength of weak ties. *American Journal of Sociology*, 78(6):1360–1380, 1973.
- [16] Yahoo! Inc. Yahoo! <http://www.yahoo.com/>.
- [17] Google. Inc. Google. <http://www.google.com/>.
- [18] ETS. The test of english as a foreign language (toefl). <http://www.toefl.org/>.
- [19] O Williamson. Market and hierarchies. *New York: Free Press*, 1975.
- [20] 独立行政法人 労働政策研究・研修機構. 労働政策研究支援情報. <http://db.jil.go.jp/>.
- [21] リクルート ワークス研究所. <http://www.works-i.com/>.
- [22] International Business Machines Corporation. Lotus notes. <http://www.lotus.com/>.
- [23] Microsoft Corporation. Microsoft exchange server. <http://www.microsoft.com/exchange/>.
- [24] サイボウズ株式会社. サイボウズ office. <http://cybozu.co.jp/products/>.
- [25] 株式会社ネオジャパン. ioffice. <http://www.neo.co.jp/ioffice/ioffice/product/>.
- [26] eGroups KK. Yahoo! eグループ. <http://www.egroups.co.jp/>.
- [27] 国立情報学研究所 and 株式会社NTT データポケット. Netcommons. <http://www.net-commons.org/wp/>.
- [28] About. Inc. About. <http://www.about.com/>.
- [29] 株式会社オーケイウェブ. Okweb. <http://www.okweb.ne.jp/>.
- [30] 株式会社ドウ・ハウス. ひとびと?net. <http://www.hitobito.net/>.
- [31] リアルコム株式会社. Kスクエア. <http://www.ksquare.co.jp/>.
- [32] Groove Networks. Inc. Groove workspace. <http://www.groove.net/>.
- [33] アリエル・ネットワーク株式会社. Ariel airone. <http://www.ariel-networks.com/product/>.
- [34] Microsoft Corporation. Msn messenger. <http://messenger.msn.com/>.
- [35] America Online. Inc. Aol messenger. <http://www.aol.com/aim/>.
- [36] Bolt. Inc. Bolt. <http://www.bolt.com/>.
- [37] エキサイト株式会社. エキサイトフレンズ. <http://friends.excite.co.jp/friends/>.

- [38] Sports Match Online. Inc. Sports match online. <http://www.sportsmatchonline.com/>.
- [39] D Koukka and L Harada. Integrating information via matchmaking. *Journal of Information Systems*, O:101–121, 1996.
- [40] F Foner. A multi-agent referral system for matchmaking. In *Proceedings of the First International Conference on Practical Application of the Intelligent Agents and Multi-Agent Technology (PAAM '96)*, 1996.
- [41] Toyooki Nishida, Takashi Hirata, and Harumi Maeda. Comemo-community: A system for supporting community knowledge evolution.
- [42] Microsoft Corporation. Advanced networking pack for windows xp. <http://support.microsoft.com/default.aspx?scid=kb>
- [43] Microsoft Corporation. Three degrees. <http://threedegrees.com/>.
- [44] Napster LLC. Napster music community. <http://www.napster.com/>.
- [45] Gnutella. <http://gnutella.wego.com>.
- [46] 高橋 範泰 and 山下 剛史. 知人のネットワークの概念に基づいた情報共有機構. 情報処理学会研究報告, 98(65), 1998.
- [47] N Contractor, D Zink, and M Chan. Iknow: A tool to assist and study the creation, maintenance, and dissolution of knowledge networks. *Community Computing and Support Systems*, LNCS1519, 1999.
- [48] 株式会社ユニークアイディ. 関心空間. <http://www.kanshin.com/>.
- [49] 有限会社 はてな. はてなダイアリー. <http://d.hatena.ne.jp/>.
- [50] Friendster. Inc. friendster. <http://www.friendster.com/>.
- [51] Tickle Inc. Tickle american's social network. <http://connect.tickle.com/>.
- [52] Stanford University. Incircle. <http://incircle.stanfordalumni.org/>.
- [53] W. Yeong, T. Howes, and S. Kille. *Lightweight Directory Access Protocol*, March 1995. RFC 1777.
- [54] K. Zeilenga. *Lightweight Directory Access Protocol version 2 (LDAPv2) to Historic Status*, March 2003. RFC 3494.
- [55] ICQ Inc. Icq. <http://web.icq.com/>.
- [56] Jabber. Inc. Jabber. <http://www.jabber.org/>.

- [57] the foaf project. Foaf. <http://www.foaf-project.org/>.
- [58] Lassila Ora et al. *Resource Description Framework (RDF) Model and Syntax Specification*. W3C Recommendation, 1999. <http://www.w3.org/TR/REC-rdf-syntax>.
- [59] Dan Brickley and R.V Guha. *RDF Vocabulary Description Language 1.0: RDF Schema*. W3C Working Draft, 2002. <http://www.w3.org/TR/rdf-schema>.
- [60] Dave Beckett. *RDF/XML Syntax Specification (Revised)*. W3C Working Draft, 2002. <http://www.w3.org/TR/rdf-syntax-grammar/>.
- [61] Sun Microsystems. *Java 2 Platform API Specification (JDK1.4)*. Sun Microsystems, 2003. <http://java.sun.com/j2se/1.4.2/docs/api/index.html>.
- [62] Sun Microsystems. *J2EE Platform Specification (1.4 - Final Release)*. Sun Microsystems, 2003. <http://java.sun.com/j2ee/>.
- [63] Jakarta Project. *Struts*. Apache Software Foundation, 2003. <http://jakarta.apache.org/struts/>.
- [64] Sun Microsystems. *JSP Standard Tag Library*. Sun Microsystems, 2003. <http://java.sun.com/products/jsp/jstl/>.
- [65] The PostgreSQL Global Development Group. *PostgreSQL*. The PostgreSQL Global Development Group. <http://www.postgresql.org/>.
- [66] Sun Microsystems. *Java Naming and Directory Interface (JNDI)*. Sun Microsystems. Java Naming and Directory Interface (JNDI).
- [67] 須子 善彦, 宮川 祥子, and 折田 明子. 自己情報コントロール権を実現した人材マッチングシステムの研究. 情報処理学会 研究報告「グループウェアとネットワークサービス」, No.044(001), 2002.
- [68] Yoshihiko Suko, Haruki Yokoyama, Shingo Yamada, Yuzuru Takeuchi, Ken'ichi Shimada, Takashi Nagano, Kenji Koizumi, and Fumitoshi Kato. The demonstration using real time location & profile data for basic environment of gaming & simulation system matching game using rfid tag and mobile phone. *In Proceedings of International Simulation And Gaming Association (ISAGA) The 34th Annual Conference*, No.117:1223–1232, 2003.
- [69] THE BOX Project. Report of the box project. <http://isaga.sfc-connect.net/>.
- [70] 慶應義塾大学 SFC Incubation Village 研究コンソーシアム. <http://www.siv.ne.jp/>.
- [71] Albert-László Barabási. *Linked: The New Science of Networks*. Perseus Publishing, 2002.